

The Land Data Toolkit (LDT) User's Guide

October 22, 2014

Revision 1.1

History:

Revision	Summary of Changes	Date
1.1	LDT Public Release patch 1	October 22, 2014



National Aeronautics and Space Administration
Goddard Space Flight Center
Greenbelt, Maryland 20771

History:

Revision	Summary of Changes	Date
1.0	Initial Version for LDT Documentation	July 8, 2014

Contents

1	Introduction	5
1.1	What's New	5
1.1.1	Version 7.0	5
2	Background	6
2.1	LDT	6
2.2	Summary of key features	6
3	Preliminary Information	8
4	Obtaining the Source Code	9
4.1	Public Release Source Code Tar File	9
4.2	Checking Out the Source Code	9
5	Building the Executable	11
5.1	Development Tools	11
5.2	Required Software Libraries	11
5.3	Optional Software Libraries	12
5.4	Build Instructions	13
5.5	Generating documentation	16
6	Running the Executable	17
7	LDT config File	18
7.1	Overall driver options	18
7.2	Domain specification	25
7.2.1	Cylindrical lat/lon	26
7.2.2	Lambert conformal	26
7.2.3	Gaussian	27
7.2.4	Polar stereographic	27
7.2.5	HRAP	27
7.2.6	Mercator	28
7.3	Parameters	28
7.4	Crop-Irrigation Parameters	32
7.5	Soil Parameters	35
7.6	Topography Parameters	38
7.7	LSM-specific Parameters	41
7.8	Climate Parameters	55
7.9	Forcing Parameters	57
7.9.1	NLDAS-2 Forcing based parameter inputs	57
7.9.2	NLDAS-1 Forcing based parameter inputs	57
7.9.3	PRINCETON Forcing based parameter inputs	57
7.9.4	NAM242 Forcing based parameter inputs	58
7.9.5	GDAS	58
7.9.6	ECMWF	58

7.10 Ensemble restart model options	59
7.11 Data Assimilation preprocessing options	61
8 Configuration of parameter attributes	69
A Description of output files from LDT	74
A.0.1 Dimensions attributes	74
A.0.2 Variable attributes	74
B Cylindrical Lat/Lon Domain Example	76
C Lambert Conformal Domain Example	81
D Gaussian Domain Example	82
E Polar Stereographic Domain Example	86
F HRAP Domain Example	87
G Mercator Domain Example	88

1 Introduction

This is the User's Guide for the Land surface Data Toolkit (LDT). This document describes how to download and install the LDT software and instructions on building an executable.

This document consists of several sections, described as follows:

- 1 Introduction:** the section you are currently reading
- 2 Background:** general information about the LDT
- 3 Preliminary Information:** general information, steps, instructions, and definitions used throughout the rest of this document
- 4 Obtaining the Source Code:** the steps needed to download the source code
- 5 Building the Executable:** the steps needed to build the LDT executable

1.1 What's New

1.1.1 Version 7.0

1. This is the initial version developed for processing data inputs to LIS version 7.0 or higher.

2 Background

A key step in preparing for land surface model (LSM) and hydrologic model simulations is ensuring that all parameters and data inputs belong to the same grid, projection, units, etc. The Land surface Data Toolkit (LDT) is an integrated framework designed specifically for processing data inputs for such land surface and hydrological models. The system not only acts as a pre-processor to the NASA Land Information System (LIS), which is an integrated framework designed for multi-model LSM simulations and data assimilation (DA) integrations, but as a land surface-based observation and DA input processor. LDT is also capable of deriving restart inputs and ensuring data quality control for inputs to LSMs and DA routines.

2.1 LDT

LDT provides an environment for processing LSM data and parameters, restart files and data assimilation based inputs (e.g., for bias correction methods). LDT offers and will offer a variety of user options and inputs to processing datasets for use within LIS and even stand-alone models. LDT is being designed with not only LIS in mind but for other independent models and data processing systems as well. This intended design is facilitated by the use of common data formats, like NetCDF, which provide detailed data header information.

LDT shares similar object oriented framework designs as LIS, with a number of points of flexibility known as “plugins”. Specific implementations are added to the framework through the plugin-interfaces. LDT uses the plugin-based architecture to support the processing of different types of observational data sets, ranging from in-situ, satellite and remotely sensed and reanalysis products.

2.2 Summary of key features

The key capabilities of LDT can be summarized as follows:

- Processing and grouping parameters needed for different LSMs and hydrologic models
- Producing observation-based data assimilation inputs (e.g., CDF matching)
- Generate custom-made restart files for LSMs
- Read in a variety of model inputs, for example:
 - Land cover maps — UMD AVHRR, MODIS-IGBP, USGS, etc.
 - Soil parameters — soil fractions, texture, color, etc.
 - Topographic — elevation, slope, aspect, etc.
 - Dynamic parameters — greenness fraction, LAI/SAI, albedo, etc.

- Expanding LSM parameter tiling options to include topographic, soils, and other parameter information, beyond just land cover type tiling
- Options for reading in or deriving a land/water mask during processing

3 Preliminary Information

This section provides some preliminary information to make reading this guide easier.

Commands are written like this:

```
% cd /path/to/LDT  
% ls  
“... compiler flags, then run gmake.”
```

File names are written like this:

/path/to/LDT/src

You need to create a working directory on your system to install LDT.

Let's call this directory */path/to/LDT/*. Throughout the rest of this document, this directory shall be referred to as *\$WORKING*. You should create a directory to run LDT in. This directory can be created anywhere on your system, but, in this document, we shall refer to this running directory as *\$WORKING/run*.

4 Obtaining the Source Code

This section describes how to obtain the source code needed to build the LDT executable.

Due to the history of LDT's development, the source code may not be freely distributed. The source code is available only to U.S. government agencies or entities with a U.S. government grant/contract. LDT's web-site explains how qualified persons may request a copy of the source code.

4.1 Public Release Source Code Tar File

After the proper paperwork and Software Usage Agreement have been completed, the qualified person will be given an official LDT revision 7.0 tar-file containing the source code.

1. Create a directory to unpack the tar-file into. Let's call it *TOPLEVELDIR*.
2. Place the tar-file in this directory.
`% mv LDT_public_release_7.0r.tar.gz TOPLEVELDIR`
3. Go into this directory.
`% cd TOPLEVELDIR`
4. Run `gzip -dc LDT_public_release_7.0r.tar.gz | tar xf -`
This command will unzip and untar the tar-file.

Note that the directory containing the LDT source code will be referred to as *\$WORKING* throughout the rest of this document.

4.2 Checking Out the Source Code

The source code is maintained in a Subversion repository. Only developers may have access to the repository. Developers must use the Subversion client (`svn`) to obtain the LDT source code. If you need any help regarding Subversion, please go to <http://subversion.apache.org/>.

Developers must first obtain access to the LDT source code repository. To obtain access you must contact the LDT team. Once you have access to the repository, you will be given the correct Subversion command to run to checkout the source code.

1. Create a directory to checkout the code into. Let's call it *TOPLEVELDIR*.
2. Go into this directory.
`% cd TOPLEVELDIR`
3. Check out the source code into a directory called *src*.
For the public version, run the following command:
`% svn checkout https://progress.nccs.nasa.gov/svn/lis/7/public7.0
src`

Note that the directory containing the LDT source code will be referred to as *\$WORKING* throughout the rest of this document.

Source code documentation may be found on LDT's web-site. Follow the "Documentation" link.

5 Building the Executable

This section describes how to build the source code and create LDT' executable — named LDT.

5.1 Development Tools

This code has been compiled and run on Linux PC (Intel/AMD based) systems, IBM AIX systems, and SGI Altix systems. These instructions expect that you are using such a system. In particular you need:

- Linux
 - Compilers
 - * either Intel Fortran Compiler version 11 or version 13 with corresponding Intel C Compiler
 - * or Absoft's Pro Fortran Software Developement Kit, version 10.0 with GNU's C and C++ compilers, gcc and g++, version 3.3.3
 - * or Lahey/Fujitsu's Fortran 95 Compiler, release L6.00c with GNU's C and C++ compilers, gcc and g++, version 3.3.3
 - GNU's make, gmake, version 3.77
- IBM
 - XL Fortran version 10.1.0.6
 - GNU's make, gmake, version 3.77
- SGI Altix
 - Intel Fortran Compiler version 12
 - GNU's make, gmake, version 3.77

5.2 Required Software Libraries

In order to build the LDT executable, the following libraries must be installed on your system:

- Earth System Modeling Framework (ESMF) version 5.2.0rp3 (or higher). (<http://www.earthsystemmodeling.org/download/releases.shtml>)
Note that starting with ESMF version 5, the ESMF development team is trying to maintain backwards compatibility with its subsequent releases. The LDT development team, however, has neither compiled nor tested against versions of ESMF newer than 5.2.0rp3.
- GRIB-API version 1.9.16
(<https://software.ecmwf.int/wiki/display/GRIB/Home>)
Note that GRIB-API requires the JasPer library (<http://www.ece.uvic.ca/frodo/jasper/>).

- NetCDF either version 3.6.3 or version 4.3.0 (or higher).
[\(http://www.unidata.ucar.edu/software/netcdf/\)](http://www.unidata.ucar.edu/software/netcdf/)

Please read the on-line documentation for details on installing NetCDF.

Addtional notes for NetCDF 4:

- You must also choose whether to compile with compression enabled. Compiling with compression enabled requires HDF 5 and zlib libraries. To enable compression, add `--enable-netcdf-4` to the `configure` options. To disable compression, add `--disable-netcdf-4` to the `configure` options.

An example of installing NetCDF 4 without compression:

```
% ./configure --prefix=$HOME/local/netcdf-4.3.0 --disable-netcdf-4
% gmake
% gmake install
```

An example of installing NetCDF 4 with compression:

```
% CPPFLAGS=-I$HOME/local/hdf5/1.8.11/include \
> LDFLAGS=-L$HOME/local/hdf5/1.8.11/lib \
> ./configure --prefix=$HOME/local/netcdf/4.3.0 --enable-netcdf-4
% gmake
% gmake install
```

- You must also download the *netcdf-fortran-4.2.tar.gz* file. First install the NetCDF C library, then install the NetCDF Fortran library. Again, please read the on-line documentation for more details.

An example of installing the NetCDF 4 Fortran library:

```
% LD_LIBRARY_PATH=$HOME/local/netcdf/4.3.0/lib:$LD_LIBRARY_PATH \
> CPPFLAGS=-I$HOME/local/netcdf/4.3.0/include \
> LDFLAGS=-L$HOME/local/netcdf/4.3.0/lib \
> ./configure --prefix=$HOME/local/netcdf/4.3.0
% gmake
% gmake install
```

5.3 Optional Software Libraries

The following libraries are not required to compile LDT. They are used to extend the functionality of LDT.

- HDF

You may choose either HDF version 4, HDF version 5, or both.

HDF is used to support a number of remote sensing datasets.

If you wish to use MODIS snow cover area observations or NASA AMSR-E soil moisture observations, then you need HDF 4 support.

If you wish to use ANSA snow cover fraction observations, then you need HDF 5 support.

If you wish to use PMW snow observations, then you need both HDF 4 and HDF 5 support.

– HDF 4

If you choose to have HDF version 4 support, please download the HDF source for version 4.2r4 from (<http://www.hdfgroup.org/products/hdf4>) and compile the source to generate the HDF library. Make sure that you configure the build process to include the Fortran interfaces.

Note that when compiling LDT with HDF 4 support, you must also download and compile HDF-EOS2 (<http://hdfeos.org/>).

– HDF 5

If you choose to have HDF version 5 support, please download the HDF source for version 1.8.11 from (<http://www.hdfgroup.org/HDF5/>) and compile the source to generate the HDF library. Make sure that you configure the build process to include the Fortran interfaces.

To install these libraries, follow the instructions provided at the various URL listed above. These optional libraries have their own dependencies, which should be documented in their respective documentation.

Note that due to an issue involving multiple definitions within the NetCDF 3 and HDF 4 libraries, you cannot compile LDT with support for both NetCDF 3 and HDF 4 together.

Note that due to the mix of programming languages (Fortran and C) used by LDT, you may run into linking errors when building the LDT executable. This is often due to (1) the Fortran compiler and the C compiler using different cases (upper case vs. lower case) for external names, and (2) the Fortran compiler and C compiler using a different number of underscores for external names.

When compiling code using Absoft's Pro Fortran SDK, set the following compiler options:

`-YEXT_NAMES=LCS -s -YEXT_SFX=_ -YCFRL=1`

These must be set for each of the above libraries.

5.4 Build Instructions

1. Perform the steps described in Section 4 to obtain the source code.
2. Goto the `$WORKING` directory. This directory contains two scripts for building the LDT executable: `configure` and `compile`.
3. Set the `LDT_ARCH` environment variable based on the system you are using. The following commands are written using Bash shell syntax.
 - For an AIX system
`% export LDT_ARCH=AIX`

- For a Linux system with the Intel Fortran compiler
`% export LDT_ARCH=linux_ifc`
- For a Linux system with the Absoft Fortran compiler
`% export LDT_ARCH=linux_absoft`
- For a Linux system with the Lahey Fortran compiler
`% export LDT_ARCH=linux_lf95`

It is suggested that you place this command in your *.profile* (or equivalent) startup file.

4. Run the *configure* script first by typing:

```
% ./configure
```

This script will prompt the user with a series of questions regarding support to compile into LDT, requiring the user to specify the locations of the required and optional libraries via several LDT specific environment variables. The following environment variables are used by LDT.

Variable	Description	Usage
LDT_FC	Fortran 90 compiler	required
LDT_CC	C compiler	required
LDT_MODESMF	path to ESMF module files	required
LDT_LIBESMF	path to ESMF library files	required
LDT_JASPER	path to JasPer library	required
LDT_GRIBAPI	path to GRIB-API library	required
LDT_NETCDF	path to NetCDF library	required
LDT_HDF4	path to HDF4 library	optional
LDT_HDF5	path to HDF5 library	optional
LDT_HDFEOS	path to HDFEOS library	optional

It is suggested that you add these definitions to your *.profile* (or equivalent) startup file.

Note that the CC variable must be set to a C compiler, not a C++ compiler. A C++ compiler may mangle internal names in a manner that is not consistent with the Fortran compiler. This will cause errors during linking.

5. An example execution of the *configure* script is shown below:

```
% ./configure
-----
Setting up configuration for LDT version 7.0...
Optimization level (-2=strict checks, -1=g, 0,1,2,3, default=2):
Assume little/big_endian data format (1-little, 2-big, default=2):
Use openMP parallel dtm (1=yes, 0-no, default=0):
NETCDF version (3 or 4, default=4)?:
NETCDF use shuffle filter? (default = 1):
```

```

NETCDF use deflate filter? (default = 1):
NETCDF use deflate level? (default = 9):
Use HDF4? (1=yes, 0=no, default=1):
Use HDF5? (1=yes, 0=no, default=1):
Use HDFEOS? (1=yes, 0=no, default=1):
Include date/time stamp history? (1=yes, 0=no, default=1):
-----
configure.ldt file generated successfully
-----
Settings are written to configure.ldt in the make directory
If you wish to change settings, please edit that file.
To compile, run the compile script.
-----
```

At each prompt, select the desired value. If you desire the default value, then you may simply press the Enter key.

Most of the configure options are be self-explanatory. Here are a few specific notes:

- for `Assume little/big_endian data format (1-little, 2-big, default=2):`, select the default value of 2. By default, LDT reads and writes binary data in the big endian format. Only select the value of 1, if you have reformatted all required binary data into the little endian format.
- for `Use GRIBAPI? (1=yes, 0=no, default=1):`, select the default value of 1. Technically, GRIB support is not required by LDT; however, the most of the commonly used met forcing data are in GRIB, making GRIB support a practical requirement.

Note that due to an issue involving multiple definitions within the NetCDF 3 and HDF 4 libraries, you cannot compile LDT with support for both NetCDF 3 and HDF 4 together.

Note that if you compiled NetCDF 4 without compression, then when specifying `NETCDF version (3 or 4, default=4):`, select 3. Then you must manually append `-lncdff` to the `LDLFLAGS` variable in the *make/configure.ldt* file.

6. Compile the LDT source code by running the *compile* script.

```
% ./compile
```

This script will compile the libraries provided with LDT, the dependency generator and then the LDT source code. The executable *LDT* will be placed in the `$WORKING` directory upon successful completion of the *compile* script.

7. Finally, copy the *LDT* executable into your running directory, `$RUNNING`.

5.5 Generating documentation

LDT code uses the ProTex (<http://gmao.gsfc.nasa.gov/software/protex/>) documenting system [1]. The documentation in L^AT_EX format can be produced by using the `doc.csh` in the `$WORKING/utils` directory. This command produces documentation, generating a number of L^AT_EX files. These files can be easily converted to pdf using utilites such as `pdflatex`.

6 Running the Executable

This section describes how to run the LDT executable.

First you should create a directory to run LDT in. It is suggested that you run LDT in a directory that is separate from your source code. This running directory shall be referred to as *\$RUNNING*. Next, copy the LDT executable into your running directory.

```
% cp $WORKING/LDT $RUNNING
```

The single-process version of LDT is executed by the following command issued in the *\$RUNNING* directory.

```
% ./LDT
```

Note that when using the Lahey Fortran compiler, you must issue this command to run the single-process version of LDT:

```
% ./LDT -Wl,T
```

Some systems require that you submit your job into a batch queue. Please consult with your system administrator for instructions on how to do this.

Note that before running LDT, you must set your environment to have an unlimited stack size. For the Bash shell, run

```
% ulimit -s unlimited
```

To customize your run, you must modify the *ldt.config* configuration file. See Section 7 for more information.

7 LDT config File

This section describes the options in the *ldt.config* file.

Not all options described here are available in the public version of LDT.

7.1 Overall driver options

LDT Running mode: specifies the running mode used in LDT. Acceptable values are:

Value	Description
“LSM parameter processing”	LSM Parameter Processing Option
“DA preprocessing”	Data Assimilation Preprocessing Option
“Ensemble restart processing”	Deriving an ensemble restart file Option
“Restart preprocessing”	LSM Restart File Preprocessing Option

LDT running mode:	"LSM parameter processing"
--------------------------	----------------------------

Parameter type list file: specifies the attribute list of LSM parameter file types to be processed within LDT and then provided to LIS. See the sample *param_attribs.txt* (Section 8) file for a complete specification description.

LSM parameter attributes file:	./param_attribs.txt
---------------------------------------	---------------------

Processed LSM parameters file: specifies the output filename (with netcdf extension) of the LSM parameters processed in LDT to go into LIS. See a sample *lis_input.d01.nc* (Appendix A) file for a complete specification description.

Processed LSM parameter filename:	./lis_input.d01.nc
--	--------------------

LIS number of nests: specifies the number of nests used for the run. Values 1 or higher are acceptable. The maximum number of nests is limited by the amount of available memory on the system. The specifications for different nests are done using white spaces as the delimiter. Please see below for further explanations. Note that all nested domains should run on the same projection and same land surface model.

LIS number of nests:	1
----------------------	---

Number of surface model types: specifies the number of surface model types selected for the LIS simulation. Acceptable values are 1 or higher.

Number of surface model types: 1

Surface model types: specifies the names of the surface model types. Options include (but to be expanded later):

Value	Description
LSM	Land surface model type
Openwater	Openwater surface type

Surface model types: "LSM"

Land surface model: specifies the land surface model to be run. Acceptable values are:

Value	Description
none	Template LSM
Noah.2.7.1	Noah 2.7.1
Noah.3.2	Noah 3.2
Noah.3.3	Noah 3.3
CLM.2	CLM version 2.0
VIC.4.1.1	VIC 4.1.1
VIC.4.1.2	VIC 4.1.2
Mosaic	Mosaic
HySSIB	HySSIB
CLSMF2.5	Catchment, Fortuna 2.5
SAC.3.5.6	Sacramento
SNOW17	Snow17
RDHM.3.5.6	Sacramento+snow17
GeoWRSI.2	GeoWRSI, v2.0

Land surface model: Noah.3.3

Lake model: specifies the lake model type used in a LIS run. Currently, only the FLake lake model is incorporated in LIS, and both LDT and LIS are set up for additional support of lake model installation and development. For now, the option “none” is recommended.

Lake model:	none
--------------------	------

Water fraction cutoff value: specifies what gridcell fraction is to be represented by water (e.g., 0.6 would be 60% covered by water pixels). This value acts as a threshold in determining which gridcell will be included as a water or land point (used also in deriving the land/water mask).

Water fraction cutoff value:	0.5
-------------------------------------	-----

Number of met forcing sources: specifies the number of met forcing datasets to be used. Acceptable values are 0 or higher.

Number of met forcing sources:	1
---------------------------------------	---

Met forcing sources: specifies the meteorological forcing data sources used for a LIS run. Acceptable values for the sources are:

Value	Description
“NONE”	none
“CMAP”	CMAP precipitation fields
“CPC_CMORPH”	CMORPH precipitation fields
“ECMWF”	ECMWF near-realtime analysis
“ECMWF_reanalysis”	ECMWF reanalysis(II), based on Berg et al.(2003)
“GDAS”	GDAS near-realtime analysis
“GEOS”	NASA-GEOS (v3-5) forcing analysis
“GEOS5_forecast”	GEOS v5 forecast fields
“GFS”	NCEP-GFS forecast fields
“GLDAS”	Coarse-scale GLDAS-1 forcing
“GSWP1”	GSWP1 forcing
“GSWP2”	GSWP2 forcing
“MERRA_Land”	NASA’s MERRA-Land reanalysis
“NAM242”	NCEP-NAM 242 resolution (Alaska)
“NARR”	North American Regional Reanalysis
“NLDAS1”	NLDAS1 analysis fields
“NLDAS2”	NLDAS2 analysis fields
“PRINCETON”	Global Princeton long-term forcing records
“RFE2(daily)”	CPC Daily Rainfall estimator fields
“RFE2(gdas)”	CPC RFE2 rainfall adjusted with GDAS/CMAP precipitation
“CPC_STAGEII”	CPC Stage II radar-based rainfall
“CPC_STAGEIV”	CPC Stage IV radar-based rainfall
“TRMM_3B42RT”	TRMM-based 3B42 real-time rainfall
“TRMM_3B42V6”	TRMM-based 3B42 V6 rainfall
“TRMM_3B42V7”	TRMM-based 3B42 V7 rainfall

Met forcing sources:	"NLDAS2"
----------------------	----------

Met spatial transform methods: specifies the type of spatial transform or interpolation scheme to apply to the forcing dataset(s) selected. Acceptable values are:

Value	Description
“average”	Upscale via averaging
“neighbor”	Nearest neighbor scheme
“bilinear”	Bilinear interpolation scheme
“budget-bilinear”	Conservative interpolation scheme (“conserves” quantities)

Bilinear interpolation uses 4 neighboring points to compute the interpolation weights. The conservative approach uses 25 neighboring points. This option is designed to conserve water, like for precipitation. Also, nearest neighbor can be used, which may better preserve large pixels (e.g., 1x1 deg), if needed. “Average” can also be selected if upscaling from finer-scale meteorological fields

(e.g., going from 4 KM to 0.25 deg).

Met spatial transform methods:	bilinear
--------------------------------	----------

Topographic correction method (met forcing): specifies whether to use elevation correction on select forcing fields. Acceptable values are:

Value	Description
"none"	Do not apply topographic correction for forcing
"lapse-rate"	Use lapse rate correction for forcing

Current meteorological forcing datasets supported for applying this lapse-rate adjustment to the temperature, humidity, pressure and downward longwave fields, include: NLDAS1, NLDAS2, NAM242, GDAS, PRINCETON, and ECMWF Future forcing dataset options will include: GEOS, GLDAS, GFS, ECMWF_reanalysis, and possible others.

ECMWF and GDAS forcing types include several terrain height maps and not just one overall, either due to change in versions or gridcell size, respective.

Topographic correction method (met forcing):	"lapse-rate"
--	--------------

LDT diagnostic output file: specifies the name of run time LDT diagnostic file.

LDT diagnostic file:	ldtlog
----------------------	--------

Metrics output directory: specifies the directory name for writing out metrics information Acceptable values are any 40 character string. The default value is set to OUTPUT.

Metrics output directory:	OUTPUT
---------------------------	--------

Undefined value: specifies the undefined value. The default is set to -9999.

Undefined value:	-9999.0
------------------	---------

Number of ensembles per tile: specifies the number of ensembles of tiles to be used. The value should be greater than or equal to 1.

Number of ensembles per tile:	1
-------------------------------	---

The following options are used for subgrid tiling based on vegetation or other parameter types (e.g., soil type, elevation, etc.), and are required for generating an ensemble restart file or downscaling to a single-member restart file from an ensemble one. See the section on ensemble restart files for more information.

Maximum number of surface type tiles per grid: defines the maximum surface type tiles per grid (this can be as many as the total number of vegetation types). Note: Allowable values are greater than or equal to 1. Note that the entry here should exactly match the entry used in the lis.config file.

Maximum number of surface type tiles per grid: 1
--

Minimum cutoff percentage (surface type tiles): defines the smallest percentage of a cell for which to create a tile. The percentage value is expressed as a fraction. Allowable values are greater than or equal to 0. Note that the entry here should exactly match the entry used in the lis.config file.

Minimum cutoff percentage (surface type tiles): 0.05
--

Maximum number of soil texture tiles per grid: defines the maximum soil texture tiles per grid (this can be as many as the total number of soil texture types). Allowable values are greater than or equal to 1. Note that the entry here should exactly match the entry used in the lis.config file.

Maximum number of soil texture tiles per grid: 1
--

Minimum cutoff percentage (soil texture tiles): defines the smallest percentage of a cell for which to create a tile. The percentage value is expressed as a fraction. Allowable values are greater than or equal to 0. Note that the entry here should exactly match the entry used in the lis.config file.

Minimum cutoff percentage (soil texture tiles): 0.05
--

Maximum number of soil fraction tiles per grid: defines the maximum soil fraction tiles per grid (this can be as many as the total number of soil fraction types). Allowable values are greater than or equal to 1. Note that the entry here should exactly match the entry used in the lis.config file.

Maximum number of soil fraction tiles per grid: 1

Minimum cutoff percentage (soil fraction tiles): defines the smallest percentage of a cell for which to create a tile. The percentage value is expressed as a fraction. Allowable values are greater than or equal to 0. Note that the entry here should exactly match the entry used in the lis.config file.

Minimum cutoff percentage (soil fraction tiles): 0.05

Maximum number of elevation bands per grid: defines the maximum elevation bands per grid (this can be as many as the total number of elevation band types). Allowable values are greater than or equal to 1. Note that the entry here should exactly match the entry used in the lis.config file.

Maximum number of elevation bands per grid: 1

Minimum cutoff percentage (elevation bands): defines the smallest percentage of a cell for which to create a tile. The percentage value is expressed as a fraction. Allowable values are greater than or equal to 0. Note that the entry here should exactly match the entry used in the lis.config file.

Minimum cutoff percentage (elevation bands): 0.05

Maximum number of slope bands per grid: defines the maximum slope bands per grid (this can be as many as the total number of slope band types). Allowable values are greater than or equal to 1. Note that the entry here should exactly match the entry used in the lis.config file.

Maximum number of slope bands per grid: 1

Minimum cutoff percentage (slope bands): defines the smallest percentage of a cell for which to create a tile. The percentage value is expressed as a fraction. Allowable values are greater than or equal to 0. Note that the entry here should exactly match the entry used in the lis.config file.

Minimum cutoff percentage (slope bands): 0.05

Maximum number of aspect bands per grid: defines the maximum aspect bands per grid (this can be as many as the total number of aspect band types). Allowable values are greater than or equal to 1. Note that the entry here should exactly match the entry used in the lis.config file.

Maximum number of aspect bands per grid: 1
--

Minimum cutoff percentage (aspect bands): defines the smallest percentage of a cell for which to create a tile. The percentage value is expressed as a fraction. Allowable values are greater than or equal to 0. Note that the entry here should exactly match the entry used in the lis.config file.

Minimum cutoff percentage (aspect bands): 0.05
--

This section specifies the 2-d layout of the processors in a parallel processing environment. The user can specify the number of processors along the east-west dimension and north-south dimension using **Number of processors along x:** and **Number of processors along y:**, respectively. Note that the layout of processors should match the total number of processors used. For example, if 8 processors are used, the layout can be specified as 1x8, 2x4, 4x2, or 8x1.

Currently LDT does not have parallelization capability,

but there are future plans to incorporate this feature.

Number of processors along x: 2
Number of processors along y: 2

7.2 Domain specification

This section of the config file specifies the LIS running domain (domain over which the simulation is carried out). The specification of the LIS Run domain

section depends on the type of LIS domain and projection used. Section 7.1 lists the projections that LIS supports.

LIS Domain type: specifies the output LIS domain grid to be used with LIS. Acceptable values are:

Value	Description
latlon	Lat/Lon projection with SW to NE data ordering
lambert	Lambert conformal projection with SW to NE data ordering
gaussian	Gaussian domain
polar	Polar stereographic projection with SW to NE data ordering
HRAP	HRAP domain (based on polar stereographic projection)
mercator	Mercator projection with SW to NE data ordering

Map projection of the LIS domain: latlon
--

7.2.1 Cylindrical lat/lon

This section describes how to specify a cylindrical latitude/longitude projection. See Appendix B for more details about setting these values.

Run domain lower left lat: 25.625
Run domain lower left lon: -124.125
Run domain upper right lat: 52.875
Run domain upper right lon: -67.875
Run domain resolution (dx): 0.25
Run domain resolution (dy): 0.25

7.2.2 Lambert conformal

This section describes how to specify a Lambert conformal projection. See Appendix C for more details about setting these values.

Run domain lower left lat: 32.875
Run domain lower left lon: -104.875
Run domain true lat1: 36.875
Run domain true lat2: 36.875
Run domain standard lon: -98.875

Run domain resolution:	25.0
Run domain x-dimension size:	40
Run domain y-dimension size:	30

7.2.3 Gaussian

This section describes how to specify a Gaussian projection. See Appendix D for more details about setting these values.

Run domain first grid point lat:	-89.27665
Run domain first grid point lon:	0.0
Run domain last grid point lat:	89.27665
Run domain last grid point lon:	-0.9375
Run domain resolution dlon:	0.9375
Run domain number of lat circles:	95

7.2.4 Polar stereographic

This section describes how to specify a polar stereographic projection. See Appendix E for more details about setting these values.

Run domain lower left lat:	32.875
Run domain lower left lon:	-104.875
Run domain true lat:	36.875
Run domain standard lon:	-98.875
Run domain orientation:	0.0
Run domain resolution:	25.0
Run domain x-dimension size:	40
Run domain y-dimension size:	30

7.2.5 HRAP

This section describes how to specify a HRAP projection. See Appendix F for more details about setting these values.

Run domain lower left hrap y:	48
Run domain lower left hrap x:	17

Run domain hrap resolution:	1
Run domain x-dimension size:	1043
Run domain y-dimension size:	774

7.2.6 Mercator

This section describes how to specify a Mercator projection. See Appendix G for more details about setting these values.

Run domain lower left lat:	32.875
Run domain lower left lon:	-104.875
Run domain true lat1:	36.875
Run domain standard lon:	-98.875
Run domain resolution:	25.0
Run domain x-dimension size:	40
Run domain y-dimension size:	30

7.3 Parameters

Land cover classification: specifies the land cover classification type. Land cover or use classification types have been developed over the years by various organizations (e.g., USGS, IGBP) and research groups (e.g., satellite missions associated with groups, like UMD, BU, etc.). For more information on some of these different land classifications and their characteristics, please refer to: http://edc2.usgs.gov/glcc/globdoc2_0.php

Acceptable values are:

Value	Description
UMD	14 Landcover types
IGBP	16 Landcover types
USGS	24 Landcover types
IGBPNEP	20 Landcover types
MOSAIC	7 Landcover types
ISA	13 Landcover types
CONSTANT	2 Landcover types

Landcover classification:	"UMD"
---------------------------	-------

Landcover file: specifies the location of the vegetation classification file.

Landcover spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the landcover map. Options include:

Value	Description	Note:
none	Data is on same grid as LIS output domain	
mode	Upscale by selecting dominant type for each gridcell	
neighbor	Use nearest neighbor to select closest value	
tile	Read in tiled data or upscale by estimating gridcell fractions	
"tile"	is a special case for landcover, which allows for reading in landcover data already in tiled form, or creating tiles from finer resolution landcover data.	

```
Landcover file:          ./input/1KM/landcover_UMD.1gd4r
Landcover spatial transform:    tile
```

Landcover fill option: specifies the landcover classification data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
neighbor	Use nearest neighbor to fill missing value

Landcover fill value: indicates which landcover value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing landcover point, a value of 5 could be assigned if no nearest neighbor values exists to fill).

Landcover fill radius: specifies the radius with which to search for nearby value(s) to help fill the missing value.

```
Landcover fill option:   neighbor   # none, neighbor
Landcover fill radius:   3.         # Number of pixels to search for neighbor
Landcover fill value:   5.         # Static value to fill where missing
```

This section also outlines the domain specifications of the landcover (and for now landmask) data. If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying landcover data. Note: The Landcover grid domain inputs below are really only required by the "LIS" parameter options or files that do not include ".Native" in the parameter attributes table. All native parameters do not require the below inputs for LDT. See Appendix B for more details about setting these values.

```
Landcover map projection:      latlon
Landcover lower left lat:    -59.995
Landcover lower left lon:    -179.995
Landcover upper right lat:   89.995
Landcover upper right lon:   179.995
Landcover resolution (dx):   0.01
Landcover resolution (dy):   0.01
```

Create or readin landmask: offers the user the option to create or read in land/water mask file information. Options include the ability to impose the mask on landcover and also the other parameter fields.

```
Create or readin landmask:      "readin"
```

Landmask file: specifies the location of land/water mask file.

Note: If reading in the MOD44W land-water mask, make sure to enter 'MOD44W' in the parameter attributes table for Landmask 'source'.

```
Landmask file:      ../input/1KM/landmask_UMD.1gd4r
```

Landmask spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the landmask map. Options include:

Value	Description
none	Data is on same grid as LIS output domain
mode	Upscale by selecting dominant type for each gridcell
neighbor	Use nearest neighbor when downscaling (or upscaling, if needed)

```
Landmask spatial transform:      none
```

This section also outlines the domain specifications of the land water/mask data. The landmask map projection and extents are only needed if you specify "readin" for mask type. If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying landmask data. See Appendix B for more details about setting these values.

```
Landmask map projection:      latlon
Landmask lower left lat:     -59.995
Landmask lower left lon:     -179.995
Landmask upper right lat:    89.995
Landmask upper right lon:    179.995
Landmask resolution (dx):    0.01
Landmask resolution (dy):    0.01
```

Regional mask file: specifies the location of a regional mask file. This file can be either an index-based state, country, basin, catchment, etc. map used to mask further beyond the main water/land mask.

```
Regional mask file:  ./input/1KM/regional_statemask.1gd4r
```

Clip landmask with regional mask: A logical-based option that uses the regional mask to ‘clip’ the original landmask that is read-in or created. .true. turns on the ”clipping” option.

```
Clip landmask with regional mask: .true.
```

Regional mask spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to a regional mask map. Options include:

Value	Description
none	Data is on same grid as LIS output domain
neighbor	Use nearest neighbor to select closest value
mode	Upscale by selecting dominant type for each gridcell

```
Regional mask spatial transform: mode
```

This section also outlines the domain specifications of the regional-based land mask data. If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying regional mask data.

See Appendix B for more details about setting these values.

Regional mask map projection:	latlon
Regional mask lower left lat:	-59.995
Regional mask lower left lon:	-179.995
Regional mask upper right lat:	89.995
Regional mask upper right lon:	179.995
Regional mask resolution (dx):	0.01
Regional mask resolution (dy):	0.01

7.4 Crop-Irrigation Parameters

Incorporate crop information: specifies the logical flag with which to turn on the inclusion of crop information and also to allow the user to enter additional options that can ensure crop, landcover, and irrigation features are agreement.

Incorporate crop information: .false.

Crop classification: specifies the crop classification system used to determine the range of crops indexed for a particular crop library source.

Value	Description
none	Data is on same grid as LIS output domain
CROPMAP	19 classes; named by Ozdogan et al.(2010), used Leff et al.(2004)
FAOSTAT01	19 classes; Used by Leff et al.(2004), considered obsolete
FAOSTAT05	175 classes; Used by Monfreda et al. (2008)

Crop classification: "FAOSTAT01"

Crop library directory: specifies the source of the crop library and inventory of crop classes, related to the "Crop classification:" entry (see above).

Crop library directory: ".../LS_PARAMETERS/crop_params/Crop.Library.Files/"

Assign single crop value: specifies whether to assign a single crop value from an actual crop library inventory, such as FAOSTAT01, which is also known as the CROPMAP classification used in Ozdogan et al. (2010). By turning on

this option (.true.), you can then specify what type of crop you want to assign, like "maize", to the user-specified option, "Default crop type:". If "maize" was entered, then wherever the landcover map indicated a generic "cropland", the crop type field would be given a dominant "maize" type.

Value	Description
.false.	Do not assign a single crop class to the crop type field.
.true.	Assign a single crop type, like "maize" to the crop type field.

Assign single crop value:	.true.
---------------------------	--------

Default crop type: specifies the default crop type that the user can enter and can be used in conjunction with assigning a single crop type value (see above).

Default crop type:	"maize"
--------------------	---------

Crop type file: specifies the location of a crop type file. This file contains different crop types that can be used in conjunction with a selected land cover map (as above).

Crop type file: ./irrigation/conus_modis/UMD_N125C19.1gd4r
--

Crop map spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to a crop type map. Options include:

Value	Description	* NOTE:
none	Data is on same grid as LIS output domain	
mode	Upscale by selecting dominant type for each gridcell	
tile	Read in tiled data or upscale by estimating gridcell fractions	
LIS-7	will be expecting 'mode' or dominant crop type per gridcell at this time. Future versions will include landcover-crop tile options.	

Crop map spatial transform: mode

Irrigation type map: specifies the location of an irrigation type file. This file contains different irrigation categories (types) that can be used in conjunction with an irrigation fraction map.

A special land-use/irrigation-related map, known as the Global Rain-Fed, Irrigated, and Paddy Croplands (GRIPC) Dataset (Salmon, 2013), has also been implemented as an option to LDT. Currently, no models in LIS utilize this map but opportunities exist for the user community to utilize for their landcover and irrigation modeling needs.

```
Irrigation type map: .../LS_PARAMETERS/irrigation/irrigtype_map.bin
```

Irrigation type spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to irrigation-related maps. Options include:

Value	Description
none	Data is on same grid as LIS output domain
mode	Upscale by selecting dominant type for each gridcell
neighbor	Use nearest neighbor to select closest value
tile	Read in tiled data or upscale by estimating gridcell fractions

```
Irrigation type spatial transform: mode
```

Irrigation fraction map: specifies the location of an irrigation fraction map file. This file contains irrigation fraction (gridcell-based) that can be used in conjunction with an irrigation type map.

```
Irrigation fraction map: .../irrigation/irrig.percent.eighth.1gd4r
```

Irrigation fraction spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to irrigation-related maps. Options include:

Value	Description
none	Data is on same grid as LIS output domain
average	Upscale by averaging values for each gridcell

```
Irrigation fraction spatial transform: none
```

7.5 Soil Parameters

Soils maps

Sand fraction map: specifies the sand fraction map file.

Clay fraction map: specifies the clay fraction map file.

Silt fraction map: specifies the silt map file.

Porosity map: specifies porosity map file.

Saturated matric potential map: specifies saturated matric potential map file.

Saturated hydraulic conductivity map: specifies saturated hydraulic conductivity map file.

b parameter map: specifies b parameter map file.

Sand fraction map:	.../input/25KM/sand_FA0.1gd4r
Clay fraction map:	.../input/25KM/clay_FA0.1gd4r
Silt fraction map:	.../input/25KM/silt_FA0.1gd4r
Porosity map:	
Saturated matric potential map:	
Saturated hydraulic conductivity map:	
b parameter map:	

Soils spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the soils maps. Options include:

Value	Description
none	Data is on same grid as LIS output domain
average	Upscale by averaging values for each gridcell
neighbor	Reinterpolate by selecting nearest gridcell neighbor
bilinear	Reinterpolate by using bilinear interpolation
budget-bilinear	Reinterpolate by using conservative, budget-bilinear
tile	Read in tiled data or upscale by estimating gridcell fractions

Soils spatial transform:	average
---------------------------------	---------

Soils fill option: specifies the general soil data (e.g., fractions) fill option.

Options include:

Value	Description
none	Do not apply missing value fill routines
neighbor	Use nearest neighbor to fill missing value

By selecting the soils fill option, neighbor, this activates the need to enter values for the Soils fill radius and fill value, as shown below. If a porosity map is read in and the soils fill option is set to neighbor, the user can then enter a fill value for porosity to ensure mask-parameter agreement.

```
Soils fill option: neighbor
Soils fill radius: 3
Soils fill value: 0.33
Porosity fill value: 0.30
```

If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying soils data. See Appendix B for more details about setting these values.

```
Soils map projection: latlon
Soils lower left lat: -59.87500
Soils lower left lon: -179.87500
Soils upper right lat: 89.87500
Soils upper right lon: 179.87500
Soils resolution (dx): 0.2500
Soils resolution (dy): 0.2500
```

Soil texture map: specifies the soil texture file.

Soil texture spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the soil texture map. Options include:

Value	Description
none	Data is on same grid as LIS output domain
mode	Upscale by selecting dominant type for each gridcell
neighbor	Upscale by using nearest valid value for each gridcell
tile	Read in tiled data or upscale by estimating gridcell fractions

```
Soil texture map: ../input/25KM/soiltexture_STATSG0-FAO.1gd4r
```

```
Soil texture spatial transform:      none
```

Soil texture fill option: specifies the soil texture data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
neighbor	Use nearest neighbor to fill missing value

Soil texture fill value: indicates which soil texture value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing soil texture value, a value of 6 could be assigned if no nearest neighbor values exists to fill).

Soil texture fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

```
Soil texture fill option:      neighbor
Soil texture fill radius:      3.
Soil texture fill value:       6.
```

If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying soil texture data. See Appendix B for more details about setting these values.

```
Soil texture map projection:      latlon
Soil texture lower left lat:     -59.87500
Soil texture lower left lon:     -179.87500
Soil texture upper right lat:    89.87500
Soil texture upper right lon:    179.87500
Soil texture resolution (dx):    0.2500
Soil texture resolution (dy):    0.2500
```

Soil color map: specifies the soil color map file. This soil map is mainly used by the Community Land Model (version 2).

Soil color spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the soil color map. Options include:

Value	Description
none	Data is on same grid as LIS output domain
mode	Upscale by selecting dominant type for each gridcell
neighbor	Reinterpolate by selecting nearest gridcell neighbor

Soil color map:
Soil color spatial transform: none

If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying soil color data. See Appendix B for more details about setting these values.

Soil color map projection:	latlon
Soil color lower left lat:	-59.87500
Soil color lower left lon:	-179.87500
Soil color upper right lat:	89.87500
Soil color upper right lon:	179.87500
Soil color resolution (dx):	0.2500
Soil color resolution (dy):	0.2500

7.6 Topography Parameters

Topography maps

elevation map: specifies the elevation of the LIS grid. If the elevation map type selected is SRTM_Native, then the elevation file entry is actually just the directory path, which contains the tiled SRTM elevation files.

slope map: specifies the slope of the LIS grid. If the slope map type selected is SRTM_Native, then the file entry is actually just the directory path, which contains the tiled SRTM elevation files.

aspect map: specifies the aspect of the LIS grid. If the aspect map type selected is SRTM_Native, then the file entry is actually just the directory path, which contains the tiled SRTM elevation files.

curvature map: specifies the curvature of the LIS grid.

Elevation map:	./input/25KM/elev_GTOP030.1gd4r
Slope map:	./input/25KM/slope_GTOP030.1gd4r

Aspect map:	.../input/25KM/aspect_GTOPO30.1gd4r
Curvature map:	.../input/25KM/curv_GTOPO30.1gd4r

Elevation fill option: specifies the elevation data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
neighbor	Use nearest neighbor to fill missing value

Elevation fill value: indicates which elevation value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing elevation value, a value of 100(m) could be assigned if no nearest neighbor values exists to fill).

Elevation fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

Elevation fill option:	neighbor
Elevation fill radius:	2.
Elevation fill value:	100.

Slope fill option: specifies the slope data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
neighbor	Use nearest neighbor to fill missing value

Slope fill value: indicates which slope value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing slope value, an value of 0.1 could be assigned if no nearest neighbor values exists to fill).

Slope fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

Slope fill option:	neighbor
Slope fill radius:	2.
Slope fill value:	0.1

Aspect fill option: specifies the aspect data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
neighbor	Use nearest neighbor to fill missing value

Aspect fill value: indicates which aspect value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing aspect value, an value of 2.0 could be assigned if no nearest neighbor values exists to fill).

Aspect fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

Aspect fill option:	neighbor
Aspect fill radius:	2.
Aspect fill value:	2.0

Topography spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the topographic map. Options include:

Value	Description
none	Data is on same grid as LIS output domain
average	Upscale by averaging values for each gridcell
neighbor	Reinterpolate by selecting nearest gridcell neighbor
bilinear	Reinterpolate by using bilinear interpolation
budget-bilinear	Reinterpolate by using conservative, budget-bilinear
tile	Read in tiled data or upscale by estimating gridcell fractions

Topography spatial transform:	tile
-------------------------------	------

This section should also specify the domain specifications of the topography data. If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying topography data. See Appendix B for more details about setting these values.

Topography map projection:	latlon
Topography lower left lat:	-59.87500
Topography lower left lon:	-179.87500
Topography upper right lat:	89.87500
Topography upper right lon:	179.87500

<code>Topography resolution (dx):</code>	0.2500
<code>Topography resolution (dy):</code>	0.2500

7.7 LSM-specific Parameters

Albedo maps

Albedo map: specifies the path of the climatology based albedo files. The climatology albedo data files have the following naming convention: <directory>/<file header>.<tag>.1gd4r The tag should be either sum, win, spr, or aut depending on the season, or the tag should represent the month (such as jan, feb, mar, etc.). The file header can be anything (such as alb1KM). The albedo field is used by Noah LSM versions.

Albedo climatology interval: specifies the frequency of the albedo climatology in months.

Value	Description
monthly	Monthly interval for albedo files
quarterly	Seasonal interval for albedo files

Albedo spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the albedo maps. Options include:

Value	Description
none	Data is on same grid as LIS output domain
average	Upscale by averaging values for each gridcell
neighbor	Reinterpolate by selecting nearest gridcell neighbor
bilinear	Reinterpolate by using bilinear interpolation
budget-bilinear	Reinterpolate by using conservative, budget-bilinear

<code>Albedo map:</code>	./input/25KM/albedo_NCEP
<code>Albedo climatology interval:</code>	monthly
<code>Albedo spatial transform:</code>	none

If selecting the Catchment LSM (F2.5 version), the model requires the near infrared (NIR) and visible (VIS) albedo factor files, as shown below for example. Then select for the param.attrs.txt file: This particular albedo parameter set is currently only available for the Catchment LSM Fortuna 2.5 (CLSMF2.5).

Albedo NIR factors: 1 CLSMF2.5 - 12 1 “Description”

Albedo VIS factors: 1 CLSMF2.5 - 12 1 “Description”

These albedo parameter subroutines can be found in the albedo directory.

```
Albedo NIR factor file: ./GLDAS_1.0-deg/modis_scale_factor.albnf.clim  
Albedo VIS factor file: ./GLDAS_1.0-deg/modis_scale_factor.albvf.clim
```

Albedo fill option: specifies the albedo data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
average	Use average to fill missing value

Albedo fill value: indicates which albedo value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing albedo value, a value of 0.12 could be assigned if no nearest neighbor values exists to fill).

Albedo fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

```
Albedo fill option:           average  
Albedo fill radius:          2.  
Albedo fill value:           0.12
```

If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying albedo data. See Appendix B for more details about setting these values.

```
Albedo map projection:      latlon  
Albedo lower left lat:     -59.87500  
Albedo lower left lon:     -179.87500  
Albedo upper right lat:    89.87500  
Albedo upper right lon:    179.87500  
Albedo resolution (dx):    0.2500  
Albedo resolution (dy):    0.2500
```

Max snow albedo map: specifies the map file containing data with the static upper bound of the snow albedo. The albedo field is used by all Noah LSM and RDHM-SAC LSM versions.

Max snow albedo spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the maximum snow albedo map. Options include:

Value	Description
none	Data is on same grid as LIS output domain
average	Upscale by averaging values for each gridcell
neighbor	Reinterpolate by selecting nearest gridcell neighbor
bilinear	Reinterpolate by using bilinear interpolation
budget-bilinear	Reinterpolate by using conservative, budget-bilinear

```
Max snow albedo map:  ../input/25KM/mxsnoalb_MODIS.1gd4r
Max snow albedo spatial transform:    none
```

Max snow albedo fill option: specifies the max snow albedo data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
average	Use average to fill missing value

Max snow albedo fill value: indicates which max snow albedo value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing snow albedo value, an value of 0.42 could be assigned if no nearest neighbor values exists to fill).

Max snow albedo fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

```
Max snow albedo fill option:      average
Max snow albedo fill radius:     3.
Max snow albedo fill value:      0.42
```

If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying max snow albedo data. See Appendix B for more details about setting these values.

```
Max snow albedo map projection:      latlon
Max snow albedo lower left lat:    -59.87500
Max snow albedo lower left lon:    -179.87500
```

Max snow albedo upper right lat:	89.87500
Max snow albedo upper right lon:	179.87500
Max snow albedo resolution (dx):	0.2500
Max snow albedo resolution (dy):	0.2500

Greenness fraction maps Greenness vegetation fraction is considered the horizontal greenness fraction represented for a model gridcell. This parameter is used in the LSMs: all Noah LSMs, RDHM-SAC, Catchment F2.5.

Greenness fraction map: specifies the source of the climatology based gfrac files. The climatology greenness data files have the following naming convention: <directory>/<file header>.<tag>.1gd4r. The tag should represent the month (such as jan, feb, mar, etc.). The file header can be anything (such as green1KM).

Greenness climatology interval: specifies the frequency of the greenness climatology in months. Only current option is: “monthly”.

Calculate min-max greenness fraction: specifies a logical flag option to offer the user the ability to calculate minimum and maximum greenness fraction values from a given climatology (e.g., monthly). Acceptable values are:

Value	Description
.false.	Read in min and max greenness fraction value maps
.true.	Calculate greenness fraction from greenness climatology maps

Greenness maximum map: specifies the file of the climatological maximum greenness data from the monthly greenness files.

Greenness minimum map: specifies the file of the climatological minimum greenness data from the monthly greenness files.

Greenness spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the greenness maps. Options include:

Value	Description
none	Data is on same grid as LIS output domain
average	Upscale by averaging values for each gridcell
neighbor	Reinterpolate by selecting nearest gridcell neighbor
bilinear	Reinterpolate by using bilinear interpolation
budget-bilinear	Reinterpolate by using conservative, budget-bilinear

Greenness fraction map:	./input/25KM/gvf_NCEP
-------------------------	-----------------------

```

Greenness climatology interval: monthly
Calculate min-max greenness fraction: .true.
Greenness maximum map:      ./input/25KM/gvf_NCEP.MAX.1gd4r
Greenness minimum map:      ./input/25KM/gvf_NCEP.MIN.1gd4r
Greenness spatial transform: none

```

Greenness fill option: specifies the greenness fraction data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
average	Use average to fill missing value

Greenness fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

Greenness fill value: indicates which greenness fraction value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing greenness value, a value of 0.2 could be assigned if exists to fill).

Greenness maximum fill value: indicates which maximum greenness fraction value to be used if an arbitrary value fill is needed.

Greenness minimum fill value: indicates which minimum greenness fraction value to be used if an arbitrary value fill is needed.

```

Greenness fill option:      average
Greenness fill radius:      3
Greenness fill value:       0.20
Greenness maximum fill value: 0.80
Greenness minimum fill value: 0.05

```

If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying greenness data. See Appendix B for more details about setting these values.

```

Greenness map projection:      latlon
Greenness lower left lat:     -59.87500
Greenness lower left lon:     -179.87500
Greenness upper right lat:    89.87500
Greenness upper right lon:    179.87500

```

<code>Greenness resolution (dx):</code>	0.2500
<code>Greenness resolution (dy):</code>	0.2500

LAI/SAI maps Leaf area index and stem area index maps are used to describe the vertical representation of leafy vegetation and the woody-branch areas within a given gridcell (respectively). LAI/SAI are used in the Community Land Model (CLM), Mosaic LSM, and Catchment LSM, version F2.5.

LAI/SAI maps: specifies the source of the climatology based LAI and SAI files. The climatology data files have the following naming convention: <directory>/<file header>.<tag>.1gd4r. The tag should be represent the month (such as jan, feb, mar, etc.). The file header can be anything (such as avhrr_lai_1KM).

LAI/SAI climatology interval: specifies the frequency of the LAI or SAI climatology in months. Current option is: “monthly”.

Calculate min-max LAI: specifies a logical flag option to offer the user the ability to calculate minimum and maximum LAI values from a given climatology (e.g., monthly). Acceptable values are:

Value	Description
.false.	Read in min and max LAI value maps
.true.	Calculate LAI from LAI climatology maps

LAI maximum map: specifies the file of the climatological maximum LAI data from the monthly LAI files.

LAI minimum map: specifies the file of the climatological minimum LAI data from the monthly LAI files.

LAI/SAI spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the LAI and SAI maps. Only “none” option works for the “CLSMF2.5” source parameter entry (in the parameter attributes table). Other options for the include: “AVHRR” source parameter entry include:

Value	Description
none	Data is on same grid as LIS output domain
average	Upscale by averaging values for each gridcell
neighbor	Reinterpolate by selecting nearest gridcell neighbor
bilinear	Reinterpolate by using bilinear interpolation
budget-bilinear	Reinterpolate by using conservative, budget-bilinear

<code>LAI map:</code>	.../input/25KM/avhrr_lai_nldas
-----------------------	--------------------------------

```

SAI map:          ../input/25KM/avhrr_sai_nldas
Calculate min-max LAI:   .false.
LAI maximum map:  ../input/CLSMF2.5/clsmf2.5_maxlai.1gd4r
LAI minimum map:  ../input/CLSMF2.5/clsmf2.5_minlai.1gd4r
LAI/SAI climatology interval: monthly
LAI/SAI spatial transform:    none

```

LAI/SAI fill option: specifies the LAI/SAI data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
average	Use average to fill missing value

LAI/SAI fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

LAI fill value: indicates which LAI value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing LAI value, a value of 1 could be assigned if exists to fill).

LAI maximum fill value: indicates which maximum LAI value to be used if an arbitrary value fill is needed.

LAI minimum fill value: indicates which minimum LAI value to be used if an arbitrary value fill is needed.

SAI fill value: indicates which SAI value to be used if an arbitrary value fill is needed.

```

LAI/SAI fill option:      average
LAI/SAI fill radius:     3
LAI fill value:          1
SAI fill value:          0.5
LAI maximum fill value:  4
LAI minimum fill value:  1

```

If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying LAI/SAI data. See Appendix B for more details about setting these values.

```

LAI/SAI map projection:      latlon
LAI/SAI lower left lat:    -59.87500

```

LAI/SAI lower left lon:	-179.87500
LAI/SAI upper right lat:	89.87500
LAI/SAI upper right lon:	179.87500
LAI/SAI resolution (dx):	0.2500
LAI/SAI resolution (dy):	0.2500

Slope type map: specifies the slope type index as used in all Noah LSM versions.

Slope type spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the soils maps. Options include:

Value	Description
none	Data is on same grid as LIS output domain
mode	Upscale by selecting dominant type for each gridcell
neighbor	Use nearest neighbor to select nearest gridcell neighbor

Slope type map:	.../input/25KM/slopetype_NCEP.1gd4r
Slope type spatial transform:	none

Slope type fill option: specifies the slope type data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
neighbor	Use nearest neighbor to fill missing value

Slope type fill value: indicates which slope type value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing slope type value, an index value of 1 could be assigned if no nearest neighbor values exists to fill).

Slope type fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

Slope type fill option:	neighbor
Slope type fill radius:	2.
Slope type fill value:	1.

If the map projection of parameter data is specified to be lat/lon, the following

configuration should be used for specifying slope type data. See Appendix B for more details about setting these values.

Slope type map projection:	latlon
Slope type lower left lat:	-59.87500
Slope type lower left lon:	-179.87500
Slope type upper right lat:	89.87500
Slope type upper right lon:	179.87500
Slope type resolution (dx):	0.2500
Slope type resolution (dy):	0.2500

Bottom temperature map: specifies the bottom boundary temperature data. This parameter is currently required by the Noah LSM versions and the recently added RDHM-SAC/Snow-17 models.

Bottom temperature spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the bottom temperature map. Options include:

Value	Description
none	Data is on same grid as LIS output domain
average	Upscale by averaging values for each gridcell
neighbor	Nearest neighbor scheme
bilinear	bilinear scheme
budget-bilinear	conservative scheme

Bottom temperature map:	./input/25KM/tbot_GDAS_6YR_CLIM.1gd4r
Bottom temperature spatial transform:	none

Bottom temperature fill option: specifies the bottom boundary temperature data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
average	Averaging values for each missing value
neighbor	Use nearest neighbor to fill missing value

Bottom temperature fill value: indicates which bottom soil temperature value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing bottom temperature field, a value of 287 K could be assigned if no nearest neighbor values exists to fill).

Bottom temperature fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

```
Bottom temperature fill option: neighbor
Bottom temperature fill radius: 3.
Bottom temperature fill value: 287.0
```

Bottom temperature topographic downscaling: specifies the option with which to adjust bottom temperature field due to topographic impacts.

Value	Description
none	No topographic/elevation adjustment made to parameter data
lapse-rate	Adjust (or downscale) bottom temperature using lapse-rate correction.

```
Bottom temperature topographic downscaling: none
```

If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying bottom temperature params data. See Appendix B for more details about setting these values.

```
Bottom temperature map projection: latlon
Bottom temperature lower left lat: -59.87500
Bottom temperature lower left lon: -179.87500
Bottom temperature upper right lat: 89.87500
Bottom temperature upper right lon: 179.87500
Bottom temperature resolution (dx): 0.2500
Bottom temperature resolution (dy): 0.2500
```

Potential Evapotranspiration (PET) maps

PET directory: specifies the source of the monthly climatology based PET files. The climatology data files have the following naming convention: <directory>/<file header>.<tag>.1gd4r. The tag should be represent the month (such as JAN, FEB, MAR, etc.). The file header can be anything (such as avhrr-pet_1KM). Currently, this parameter is used only with the RDHM-SAC model.

PET adjustment factor directory: specifies the source of the m monthly climatology-based PET adjustment factor files. The climatology data files have the following naming convention: <directory>/<file header>.<tag>.1gd4r. The

tag should be represent the month (such as JAN, FEB, MAR, etc.). The file header can be anything (such as avhrr_petadj_1KM).

PET climatology interval: specifies the frequency of the PET climatology in months. Current option is: “monthly”.

PET spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to the PET maps. Options include:

Value	Description
none	Data is on same grid as LIS output domain (only option for now)

```
PET directory:          ../../input/25KM/sachtet_pet
PET adjustment factor directory: ../../input/25KM/sachtet_petadj
PET climatology interval: monthly
PET spatial transform:   none
```

PET fill option: specifies the PET climatology data fill option. Options include:

Value	Description
none	Do not apply missing value fill routines
average	Use average to fill missing value

PET fill radius: specifies the radius with which to search for nearby value(s) to help fill in the missing value.

PET fill value: indicates which PET value to be used if an arbitrary value fill is needed. (For example, when the landmask indicates a land point but no existing PET value, a value of 1 could be assigned if exists to fill.)

```
PET fill option:      average
PET fill radius:     3
PET fill value:      10.
```

If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying PET data. See Appendix B for more details about setting these values.

```
PET map projection:    latlon
PET lower left lat:   -59.87500
```

```
PET lower left lon: -179.87500
PET upper right lat: 89.87500
PET upper right lon: 179.87500
PET resolution (dx): 0.2500
PET resolution (dy): 0.2500
```

CLSMF25 tile coord file: specifies the location of a CLSM F2.5 coordinate file. This file contains catchment tile coordinate information that can be used in Catchment LSM (CLSM) Fortuna 2.5 version model run.

```
CLSMF25 tile coord file: ./cat_parms/PE_2880x1440_DE_464x224.file
```

CLSMF25 soil param file: specifies the location of a CLSM F2.5 soils file. This file contains catchment soil parameter information that can be used in Catchment LSM (CLSM) Fortuna 2.5 version model run.

```
CLSMF25 soil param file: ./cat_parms/soil_param.dat
```

CLSMF25 topo files: specifies the locations of a CLSM F2.5 topo parameter files. These files contain catchment topographic parameter information that can be used in a Catchment LSM (CLSM) Fortuna 2.5 version model run.

```
CLSMF25 topo ar file: ../cat_parms/ar.new
CLSMF25 topo bf file: ../cat_parms/bf.dat
CLSMF25 topo ts file: ../cat_parms/ts.dat
```

CLSMF25 surf layer ts file: specifies the location of a CLSM F2.5 tau parameter file. This file contain catchment surface layer timescale (ts), tau, parameter information that can be used in Catchment LSM (CLSM) Fortuna 2.5 version model runs.

```
CLSMF25 surf layer ts file: ../cat_parms/tau_param.dat
```

CLSMF25 top soil layer depth: specifies the top soil layer depth. This parameter value specifies the depth of the top soil layer depth (unit: meters) and is needed in processing other parameters for a Catchment LSM (CLSM) Fortuna 2.5 version model run.

```
CLSMF25 top soil layer depth: 0.02
```

CLSMF25 spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to CLSM F2.5 parameters. Options include (only 'none' works at this time):

Value	Description
none	Data is on same grid as LIS output domain

```
CLSMF25 spatial transform: none
```

This section also outlines the domain specifications of the Catchment LSM Fortuna 2.5 data. If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying CLSM data. See Appendix B for more details about setting these values.

```
CLSMF25 map projection: latlon
CLSMF25 lower left lat: 25.0625
CLSMF25 lower left lon: -124.9375
CLSMF25 upper right lat: 52.9375
CLSMF25 upper right lon: -67.0625
CLSMF25 resolution (dx): 0.125
CLSMF25 resolution (dy): 0.125
```

RDHM356 constants table: specifies the location of the constants table required by the Research Distributed Hydrologic Model (RDHM) version 3.5.6 models, SAC-HTET and SNOW-17. This table file contains constant values for any listed SAC-HTET or SNOW-17 parameter types. If a constant value is $\zeta=0.$, then the constant value is assigned for all gridcells for a parameter entry. If the value is negative, a 2-D gridded a priori map is read in. Also, the negative constant value can be used as a scaling factor of the 2-D grid by taking its absolute value and multiplying the entire field by it, if the value is other than -1.

RDHM356 universal undefined value: specifies an universal undefined value that can be used by either the SAC-HTET or SNOW-17 models for run-time purposes.

```
RDHM356 constants table: ./rdhm_singlevalueinputs.txt
RDHM356 universal undefined value: -1.
```

SACHTET parameter files: specifies the locations of SACHTET 3.5.6 parameter files. These files contain soil-based and other model parameter information that can be used in SAC-HTET model runs. Most parameter files will come in the HRAP domain and XMRG-binary format found commonly in NOAA NWS/OHD/RFC applications. For the soil parameters, LZ indicates "lower zone" and UZ refers to "upper zone".

SACHTET soil parameter option:	"readin"
SACHTET Cosby soil parameter table:	./testcase/cosby_eq_newzperc.txt
SACHTET soilttype parameter table:	./testcase/sachtet_soilparms.txt
SACHTET vegetation parameter table:	./testcase/sachtet_vegparms.txt
SACHTET LZFPM map:	./testcase/sac_LZFPM.gz
SACHTET LZFSM map:	./testcase/sac_LZFSM.gz
SACHTET LZPK map:	./testcase/sac_LZPK.gz
SACHTET LZSK map:	./testcase/sac_LZSK.gz
SACHTET LZTWM map:	./testcase/sac_LZTWM.gz
SACHTET UZFWM map:	./testcase/sac_UZFWM.gz
SACHTET UZTWM map:	./testcase/sac_UZTWM.gz
SACHTET UZK map:	./testcase/sac_UZK.gz
SACHTET PFREE map:	./testcase/sac_PFREE.gz
SACHTET REXP map:	./testcase/sac_REXP.gz
SACHTET ZPERC map:	./testcase/sac_ZPERC.gz
SACHTET EFC map:	./testcase/sac_EFC.gz
SACHTET PCTIM map:	./testcase/sac_PCTIM.gz
SACHTET soil albedo map:	./testcase/sachtet_soilAlbedo.gz
SACHTET offset time map:	./testcase/sachtet_offsetTime.gz
SACHTET STXT map:	./testcase/frz_STXT.gz
SACHTET TBOT map:	./testcase/frz_TBOT.gz
SACHTET CKSL map:	none
SACHTET RSMAX map:	none
SACHTET ZBOT map:	none
SACHTET parameter spatial transform:	none
SACHTET parameter fill option:	none
SACHTET map projection:	hrap

SNOW17 parameter files: specifies the locations of SNOW-17 parameter files. These files contain snow and soil-based parameter information that can be used in the SNOW-17 model run.

SNOW17 MFMAX map:	./testcase/snow_MFMAX.gz
SNOW17 MFMIN map:	./testcase/snow_MFMIN.gz
SNOW17 UADJ map:	./testcase/snow_UADJ.gz
SNOW17 ALAT map:	./testcase/snow_ALAT.gz
SNOW17 ELEV map:	./testcase/snow_ELEV.gz
SNOW17 SCF map:	none
SNOW17 NMF map:	none
SNOW17 SI map:	none
SNOW17 MBASE map:	none
SNOW17 PXTEMP map:	none
SNOW17 PLWHC map:	none
SNOW17 TIPM map:	none
SNOW17 GM map:	none
SNOW17 ELEV map:	none
SNOW17 LAEC map:	none
SNOW17 ADC directory:	none
SNOW17 parameter spatial transform:	none
SNOW17 parameter fill option:	none
SNOW17 map projection:	hrap

WRSI parameter files: specifies the locations of GeoWRSI 2.0 parameter files. These files contain crop- and soil-based parameter information that can be used in the GeoWRSI 2.0 model run.

WRSI landmask file:	./data/Africa/Static/sawmask
WRSI length of growing period file:	./data/Africa/Static/lgp_south
WRSI water holding capacity file:	./data/Africa/Static/whc3
WRSI WRSI climatology file:	./data/Africa/Static/wsimedn_edc_s
WRSI SOS climatology file:	./data/Africa/SOS/sosmedn_edc_s
WRSI SOS file:	none
WRSI SOS anomaly file:	none

7.8 Climate Parameters

Climatology parameter maps

PPT climatology maps: specifies the source of the climatology based precipitation files. The climatology precipitation data files can have the following naming conventions, depending on the data source:

PRISM: <directory>/<file header>.<tag>.txt

The file header can be anything (such as ppt_1931_2010).

The tag should represent the month (such as jan, feb, mar, etc.).

WORLDCLIM: <directory>/<file header>.<tag>.1gd4r

The file header can be prec_

The tag should represent the month (such as 1, 2,..., 12).

PPT climatology interval: specifies the frequency of the precipitation climatology in months. Current option is: “monthly“.

```
PPT climatology maps: .../LS_PARAMETERS/climate_maps/ppt_1981_2010
PPT climatology interval: monthly
```

Climate params spatial transform: indicates which spatial transform (i.e., upscale or downscale) type is to be applied to climate parameters. Only “average” spatial transform works currently for the “WORLDCLIM” climatology files. Options include:

Value	Description
none	Data is on same grid as LIS output domain
average	Upscale by averaging values for each gridcell
neighbor	Reinterpolate by selecting nearest gridcell neighbor
bilinear	Reinterpolate by using bilinear interpolation
budget-bilinear	Reinterpolate by using conservative, budget-bilinear

```
Climate params spatial transform: average
```

This section also outlines the domain specifications of climatology-based parameters, like higher scaled monthly precipitation or min/max temperatures. If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying climatology data. See Appendix B for more details about setting these values.

```
Climate params map projection: latlon
```

7.9 Forcing Parameters

7.9.1 NLDAS-2 Forcing based parameter inputs

NLDAS2 elevation difference map: specifies the NLDAS-2 elevation difference file used to remove built-in elevation correction.

NARR terrain height map: specifies the terrain height map for the NLDAS-2 base forcing of the North American Regional Reanalysis (NARR).

```
NLDAS2 elevation difference map: ../NARR_elev-diff.1gd4r  
NARR terrain height map:      ../NARR_elevation.1gd4r
```

7.9.2 NLDAS-1 Forcing based parameter inputs

NLDAS1 elevation difference map: specifies the NLDAS-1 elevation difference file used to remove built-in elevation correction.

EDAS terrain height map: specifies the terrain height map for the NLDAS-1 base forcing of the Eta Data Assimilation System (EDAS).

```
NLDAS1 elevation difference map: ../NLDAS1/EDAS_elev-diff.1gd4r  
EDAS terrain height map:      ../NLDAS1/EDAS_elevation.1gd4r
```

7.9.3 PRINCETON Forcing based parameter inputs

PRINCETON elevation map: specifies the terrain height map for the Princeton University global forcing dataset.

```
PRINCETON elevation map:      ../PRINCETON/hydro1k_elev_mean_1d.asc
```

```
NLDAS1 elevation difference map: ../NLDAS1/EDAS_elev-diff.1gd4r  
EDAS terrain height map:      ../NLDAS1/EDAS_elevation.1gd4r
```

7.9.4 NAM242 Forcing based parameter inputs

NAM242 `elevation map`:: specifies the terrain height map for the North American Mesoscale (NAM) NOAA grid 242 forcing dataset.

```
NAM242 elevation map:      ./NAM/terrain.242.grb
```

7.9.5 GDAS

GDAS parameter inputs: GDAS elevation maps specify lowest boundary layer information which can be used to downscale or lapse rate adjust GDAS meteorological variables, if given a higher resolution elevation height map. Original files are given in Grib-1 format and on their original Gaussian grids (from NCEP), so the GDAS elevation file reader is set up to support these files.

GDAS T126 `elevation map`: specifies the GDAS T126 elevation definition. GDAS T170 `elevation map`: specifies the GDAS T170 elevation definition. GDAS T254 `elevation map`: specifies the GDAS T254 elevation definition. GDAS T382 `elevation map`: specifies the GDAS T382 elevation definition. GDAS T574 `elevation map`: specifies the GDAS T574 elevation definition.

```
GDAS T126 elevation map: ./GDAS/global_orography.t126.grb
GDAS T170 elevation map: ./GDAS/global_orography.t170.grb
GDAS T254 elevation map: ./GDAS/global_orography.t254.grb
GDAS T382 elevation map: ./GDAS/global_orography.t382.grb
GDAS T574 elevation map: ./GDAS/global_orography.t574.grb
```

7.9.6 ECMWF

ECMWF parameter inputs: ECMWF elevation maps specify lowest boundary layer information which can be used to downscale or lapse rate adjust ECMWF meteorological variables, if given a higher resolution elevation height map. Original files are given in Grib-1 format and on their original lat-lon grids (from ECMWF), so the ECMWF elevation file reader is set up to support these files.

ECMWF IFS23R4 `elevation map`: specifies the ECMWF IFS23R4 terrain height map file path.

ECMWF IFS25R1 elevation map: specifies the ECMWF IFS25R1 terrain height map file path.

ECMWF IFS30R1 elevation map: specifies the ECMWF IFS30R1 terrain height map file path.

ECMWF IFS33R1 elevation map: specifies the ECMWF IFS33R1 terrain height map file path.

ECMWF IFS35R2 elevation map: specifies the ECMWF IFS35R2 terrain height map file path.

ECMWF IFS35R3 elevation map: specifies the ECMWF IFS35R3 terrain height map file path.

ECMWF IFS36R1 elevation map: specifies the ECMWF IFS36R1 terrain height map file path.

ECMWF IFS37R2 elevation map: specifies the ECMWF IFS37R2 terrain height map file path.

ECMWF IFS23R4 elevation map:	./ECMWF/ecmwf.2001092006.092006.elev_1_4
ECMWF IFS25R1 elevation map:	./ECMWF/ecmwf.2003010806.010806.elev_1_4
ECMWF IFS30R1 elevation map:	./ECMWF/ecmwf.2006020106.020106.elev_1_4
ECMWF IFS33R1 elevation map:	./ECMWF/ecmwf.2008060306.060306.elev_1_4
ECMWF IFS35R2 elevation map:	./ECMWF/ecmwf.2009031006.031006.elev_1_4
ECMWF IFS35R3 elevation map:	./ECMWF/ecmwf.2009090806.090806.elev_1_4
ECMWF IFS36R1 elevation map:	./ECMWF/ecmwf.2010012606.012606.elev_1_4
ECMWF IFS37R2 elevation map:	./ECMWF/ecmwf.2011051806.051806.elev_1_4

7.10 Ensemble restart model options

LIS restart source: specifies the surface model restart file source. Options are:

Value	Description
LSM	LSM restart file type
Routing	river or streamflow routing model restart file type

LIS restart source: "LSM"

Ensemble restart generation mode: specifies the mode of ensemble restart generation. Options are:

Value	Description
upscale	convert from a single member restart to a multi-member restart
downscale	convert from a multi-member restart to a single member restart

```
Ensemble restart generation mode: "upscale"
```

Input restart filename: specifies the name of the input restart file.

```
Input restart filename: ../OL/LIS_RST_NOAH33_201001010000.d01.nc
```

Output restart filename: specifies the name of the output restart file.

```
Output restart filename: ./LIS_RST_NOAH33_201001010000.d01.nc
```

Number of ensembles per tile (input restart): specifies the number of ensemble members used in the input restart file.

```
Number of ensembles per tile (input restart): 1
```

Number of ensembles per tile (output restart): specifies the number of ensemble members to be used in the output restart file.

```
Number of ensembles per tile (output restart): 12
```

Note: Make sure to specify the surface type, veg, soil, etc., subgrid tiling entries For upscaling or downscaling of restart files, maximum number of tiles and minimum cutoff percentage entries for subgrid tiling based on vegetation or other parameter types (e.g., soil type, elevation, etc.) are required as entries.

For example, must include, Maximum number of surface type tiles per grid:

7.11 Data Assimilation preprocessing options

The start time is specified in the following format:

Variable	Value	Description
Starting year:	integer 2001 – present	specifying starting year
Starting month:	integer 1 – 12	specifying starting month
Starting day:	integer 1 – 31	specifying starting day
Starting hour:	integer 0 – 23	specifying starting hour
Starting minute:	integer 0 – 59	specifying starting minute
Starting second:	integer 0 – 59	specifying starting second

Starting year:	2002
Starting month:	1
Starting day:	2
Starting hour:	0
Starting minute:	0
Starting second:	0

The end time is specified in the following format:

Variable	Value	Description
Ending year:	integer 2001 – present	specifying ending year
Ending month:	integer 1 – 12	specifying ending month
Ending day:	integer 1 – 31	specifying ending day
Ending hour:	integer 0 – 23	specifying ending hour
Ending minute:	integer 0 – 59	specifying ending minute
Ending second:	integer 0 – 59	specifying ending second

Ending year:	2010
Ending month:	1
Ending day:	1
Ending hour:	0
Ending minute:	0
Ending second:	0

LIS output timestep: specifies the LIS output time-step.

LIS output timestep:	1da
----------------------	-----

DA observation source: specifies the source of the observation data on which preprocessing is performed. Options are:

Value	Description
LIS LSM soil moisture	soil moisture output from a LIS run
Synthetic soil moisture	syneththic soil moisture observations created from a LIS run
AMSR-E(LPRM) soil moisture	Land Parameter Retrieval Model (LPRM) retrievals of AMSR-E soil moisture
ECV soil moisture	Essential Climate Variable (ECV) soil moisture retrievals
WindSat soil moisture	WindSat retrievals of soil moisture
RT SMOPS soil moisture	Real Time Soil Moisture Operational Processing System (SMOPS) based soil moisture retrievals
ASCAT TUW soil moisture	ASCAT soil moisture retrievals from TU Wein
GRACE TWS	Terrestrial water storage observations from GRACE

```
DA observation source: "AMSR-E(LPRM) soil moisture"
```

Apply anomaly correction to obs: specifies if observations are to be created by applying an anomaly correction. This option is only used for GRACE TWS observations.

```
Apply anomaly correction to obs: 0
```

Compute CDF: specifies if cumulative distribution function (CDF) is to be computed

```
Compute CDF: 1
```

Number of bins to use in the CDF: specifies the number of bins to use while computing the CDF.

```
Number of bins to use in the CDF: 100
```

Name of the CDF file: specifies name of the CDF file being generated. (This will be a NETCDF formatted file.)

```
Name of CDF file: "lprm_cdf"
```

Temporal averaging interval: specifies temporal averaging interval to be used while computing the CDF.

```
Temporal averaging interval: "1da"
```

Apply external mask: specifies if an external mask (time varying) is to be applied while computing the CDF.

```
Apply external mask: 0
```

External mask directory: specifies the location of the external mask.

```
External mask directory: none
```

Observation count threshold: specifies the minimum number of observations to be used for generating valid CDF data.

```
Observation count threshold: 500
```

LIS soil moisture output format: specifies the output format of the LIS model output (binary/netcdf/grib1)

LIS soil moisture output methodology: specifies the output methodology used in the LIS model output (1d tilespace/1d gridspace/2d gridspace)

LIS soil moisture output naming style: specifies the output naming style used in the LIS model output (3 level hierarchy/4 level hierarchy, etc.)

LIS soil moisture output nest index: specifies the index of the nest used in the LIS model output

LIS soil moisture output directory: specifies the location of the LIS model output

LIS soil moisture output map projection: specifies the map projection used in the LIS model output

LIS soil moisture output lower left lat: specifies the lower left latitude of the LIS model output (if map projection is latlon)

LIS soil moisture output lower left lon: specifies the lower left longitude of the LIS model output (if map projection is latlon)

LIS soil moisture output upper right lat: specifies the upper right latitude of the LIS model output (if map projection is latlon)

LIS soil moisture output upper right lon: specifies the upper right longitude of the LIS model output (if map projection is latlon)

LIS soil moisture output resolution (dx): specifies the resolution (in degrees) along the latitude of the LIS model output (if map projection is latlon)

LIS soil moisture output resolution (dy): specifies the resolution (in degrees) along the longitude of the LIS model output (if map projection is latlon)

LIS soil moisture output format:	"netcdf"
LIS soil moisture output methodology:	"2d gridspace"
LIS soil moisture output naming style:	"3 level hierarchy"
LIS soil moisture output map projection:	"latlon"
LIS soil moisture output nest index:	1
LIS soil moisture output directory:	../OL/OUTPUT
LIS soil moisture domain lower left lat:	18.375
LIS soil moisture domain lower left lon:	-111.375
LIS soil moisture domain upper right lat:	41.375
LIS soil moisture domain upper right lon:	-85.875
LIS soil moisture domain resolution (dx):	0.25
LIS soil moisture domain resolution (dy):	0.25

Synthetic soil moisture observation directory: specifies the location of the data directory containing the synthetic soil moisture data.

Synthetic soil moisture observation directory: ./SYN_SM
--

AMSR-E(LPRM) soil moisture observation directory: specifies the location of the data directory containing the LPRM AMSR-E data.

AMSR-E(LPRM) use raw data: specifies if raw data (instead of the retrievals CDF-matched to the GLDAS Noah climatology).

```
AMSR-E(LPRM) soil moisture observation directory: ./LPRM.v5  
AMSR-E(LPRM) use raw data: 1
```

ECV soil moisture observation directory: specifies the location of the data directory containing the ECV soil moisture data.

```
ECV soil moisture observation directory: ./ECV
```

WindSat soil moisture observation directory: specifies the location of the data directory containing the WindSat soil moisture data.

```
WindSat soil moisture observation directory: ./WindSat
```

RT SMOPS soil moisture observation directory: specifies the location of the data directory containing the real time SMOPS soil moisture data.

RT SMOPS soil moisture use ASCAT data: specifies if the ASCAT layer of SMOPS is to be used.

```
RT SMOPS soil moisture observation directory: ./RTSMOPS  
RT SMOPS soil moisture use ASCAT data: 1
```

ASCAT (TUW) soil moisture observation directory: specifies the location of the data directory containing the TU Wein retrievals of ASCAT soil moisture data.

```
ASCAT (TUW) soil moisture observation directory: ./TUW_ASCAT
```

GRACE raw data filename: specifies the name of the GRACE raw data.

GRACE baseline starting year: specifies the baseline starting year from which to establish the TWS climatology.

GRACE baseline ending year: specifies the baseline ending year from which to establish the TWS climatology.

LIS TWS output format: specifies the output format of the LIS model output (binary/netcdf/grib1).

LIS TWS output methodology: specifies the output methodology used in the LIS model output (1d tilespace/1d gridspace/2d gridspace).

LIS TWS output naming style: specifies the output naming style used in the LIS model output (3 level hierarchy/4 level hierarchy, etc.).

LIS TWS output nest index: specifies the index of the nest used in the LIS model output.

LIS TWS output directory: specifies the location of the LIS model output.

LIS TWS output map projection: specifies the map projection used in the LIS model output.

LIS TWS output lower left lat: specifies the lower left latitude of the LIS model output (if map projection is latlon).

LIS TWS output lower left lon: specifies the lower left longitude of the LIS model output (if map projection is latlon).

LIS TWS output upper right lat: specifies the upper right latitude of the LIS model output (if map projection is latlon).

LIS TWS output upper right lon: specifies the upper right longitude of the LIS model output (if map projection is latlon).

LIS TWS output resolution (dx): specifies the resolution (in degrees) along the latitude of the LIS model output (if map projection is latlon).

LIS TWS output resolution (dy): specifies the resolution (in degrees) along the longitude of the LIS model output (if map projection is latlon).

GRACE raw data filename:	.../GRACE_tws/GRACE.CSR.LAND.RL05.DS.G200KM.nc
GRACE baseline starting year:	2004
GRACE baseline ending year:	2009
LIS TWS output format:	"netcdf"
LIS TWS output methodology:	"2d gridspace"
LIS TWS output naming style:	"3 level hierarchy"
LIS TWS output map projection:	"latlon"

LIS TWS output nest index:	1
LIS TWS output directory:	../OL_NLDAS/OUTPUT
LIS TWS output domain lower left lat:	25.0625
LIS TWS output domain lower left lon:	-124.9375
LIS TWS output domain upper right lat:	52.9375
LIS TWS output domain upper right lon:	-67.0625
LIS TWS output domain resolution (dx):	0.125
LIS TWS output domain resolution (dy):	0.125

HYMAP river width map: specifies the name of the HYMAP river width data file.

HYMAP river height map: specifies the name of the HYMAP river height data file.

HYMAP river roughness map: specifies the name of the HYMAP river roughness data file.

HYMAP floodplain height map: specifies the name of the HYMAP floodplain height data file.

HYMAP flow direction x map: specifies the name of the x-flow direction data file

HYMAP flow direction y map: specifies the name of the y-flow direction data file

HYMAP grid elevation map: specifies the name of the grid elevation data file

HYMAP grid distance map: specifies the name of the grid distance data file

HYMAP grid area map: specifies the name of the grid area data file

HYMAP runoff time delay map: specifies the name of the runoff time delay data file.

HYMAP runoff time delay multiplier map: specifies the name of the runoff time delay multiplier data file

HYMAP baseflow time delay map: specifies the name of the baseflow time delay data file

HYMAP reference discharge map: specifies the name of the reference discharge data file

HYMAP basin mask map: specifies the name of the basin mask data file

HYMAP river width map:	./HYMAP_parms/rivwth_Getirana_Dutra.bin
HYMAP river height map:	./HYMAP_parms/rivhgt_Getirana_Dutra.bin
HYMAP river roughness map:	./HYMAP_parms/rivman_Getirana_Dutra.bin
HYMAP floodplain roughness map:	./HYMAP_parms/fldman.bin
HYMAP river length map:	./HYMAP_parms/rivlen.bin
HYMAP floodplain height map:	./HYMAP_parms/fldhgt.bin
HYMAP flow direction x map:	./HYMAP_parms/nextx.bin
HYMAP flow direction y map:	./HYMAP_parms/nexty.bin
HYMAP grid elevation map:	./HYMAP_parms/elevtn.bin
HYMAP grid distance map:	./HYMAP_parms/nxtdst.bin
HYMAP grid area map:	./HYMAP_parms/grarea.bin
HYMAP runoff time delay map:	./HYMAP_parms/kirpitch.bin
HYMAP runoff time delay multiplier map:	./HYMAP_parms/trunoff.bin
HYMAP baseflow time delay map:	./HYMAP_parms/tbasflw_45_amazon.bin
HYMAP reference discharge map:	./HYMAP_parms/qprefer.bin
HYMAP basin mask map:	./HYMAP_parms/mask_all.bin

This section also outlines the domain specifications of the HYMAP parameter data. If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying HYMAP data. See Appendix B for more details about setting these values.

HYMAP params map projection:	latlon
HYMAP params spatial transform:	none
HYMAP params lower left lat:	-59.9375
HYMAP params lower left lon:	-179.9375
HYMAP params upper right lat:	89.9375
HYMAP params upper right lon:	179.9375
HYMAP params resolution (dx):	0.125
HYMAP params resolution (dy):	0.125

8 Configuration of parameter attributes

This section defines the specification of various parameter attributes in the LDT parameter attributes file. This table file is specified in a space delimited column format. Each row consists of the following entries:

Parameter name provides the parameter or variable name, which could be included in output file. See **Source option** below.

Select option specifies whether to include the parameter or variable in the processed output file. Acceptable values are:

Value	Description
0	do not include parameter
1	include the parameter

Source option determines which data source the parameter or variable comes from or is provided by.

The appended term of _LIS or _Native to each source type indicates mainly the origin of the file and its format. If the file has _LIS attached, the file was derived originally by a LIS team member or has dimensions and format similar to a LIS type file. If the file has _Native attached to the source name, this indicates that the original or native file (not modified by the LIS team) is being read in along with its original projection, format, resolution, etc. When the _Native option is available and selected by the user, the ldt.config entries for the parameter grid file extents and resolution are not needed and LDT will ignore those entries during runtime.

Acceptable parameter names and sources include:

Names	Sources
Landcover:	AVHRR, AVHRR_GFS, MODIS_LIS, MODIS_Native,
USGS_LIS, USGS_Native, ALMIPPII, CLSMF2.5, ISA, CONSTANT	
Landmask:	AVHRR, MODIS, MOD44W, ALMIPPII, ISA
Regional mask:	ESRI, WRSI
Crop type:	UMDCROPMAP, Monfreda08, CONSTANT
Soil texture:	FAO, ZOBLER_GFS, STATSGOFAO_LIS,
STATSGOFAO_Native, STATSGOv1, CONSTANT	
Soil fractions:	FAO, STATSGOv1, CONSTANT
Soil color:	FAO
Elevation:	GTOPO30_LIS, GTOPO30_Native, GTOPO30_GFS,
SRTM_LIS, SRTM_Native, CONSTANT	
Slope:	GTOPO30, SRTM_LIS, SRTM_Native, CONSTANT
Aspect:	GTOPO30, SRTM_LIS, SRTM_Native, CONSTANT
Curvature:	GTOPO30
Slope type:	NCEP_LIS, NCEP_Native, NCEP_GFS, CONSTANT
LAI:	AVHRR, CLSMF2.5
LAImin:	CLSMF2.5
LAImax:	CLSMF2.5
SAI:	AVHRR, CLSMF2.5
Albedo:	NCEP_LIS, NCEP_Native, CONSTANT
Max snow albedo:	NCEP_LIS, NCEP_Native, NCEP_GFS,
SACHTET.3.5.6, CONSTANT	
Albedo NIR factors:	CLSMF2.5
Albedo VIS factors:	CLSMF2.5
Greenness:	NCEP_LIS, NCEP_Native, CLSMF2.5,
SACHTET.3.5.6, CONSTANT	
Shdmin:	NCEP_LIS, NCEP_Native, CONSTANT
Shdmax:	NCEP_LIS, NCEP_Native, CONSTANT
PET:	SACHTET.3.5.6
PETADJ:	SACHTET.3.5.6
Tbot:	NCEP_LIS, NCEP_GFS, ISLSCP1,
SACHTET.3.5.6, CONSTANT	
Soildepth:	ALMIPPII
Rootdepth:	ALMIPPII
Porosity:	FAO, CLSMF2.5, CONSTANT
Bedrockdepth:	CLSMF2.5
Psisat:	CLSMF2.5
Bexp:	CLSMF2.5
Ksat:	CLSMF2.5
Wpwet:	CLSMF2.5
CLSM gnu param:	CLSMF2.5
CLSM ars params:	CLSMF2.5
CLSM ara params:	CLSMF2.5
CLSM arw params:	CLSMF2.5
CLSM bf params:	CLSMF2.5
CLSM ts params:	CLSMF2.5
CLSM tau params:	CLSMF2.5

SAC-HTET:	SACHTET.3.5.6
SNOW-17:	SNOW-17
ADC:	SNOW-17
SiB2:	SiB2
WRSI LGP:	GeoWRSL.2
WRSI WHC:	GeoWRSL.2
WRSI SOSclim:	GeoWRSL.2
WRSI WRSIclim:	GeoWRSL.2
WRSI mask:	GeoWRSL.2
WRSI SOS:	GeoWRSL.2
WRSI SOSanom:	GeoWRSL.2
PPT climatology:	PRISM, WORLDCLIM
Irrig area fraction:	MODIS_OG, GRIPPC
Irrigation type:	GRIPPC
HYMAP river width:	HYMAP
HYMAP river height:	HYMAP
HYMAP river roughness:	HYMAP
HYMAP river length:	HYMAP
HYMAP floodplain height:	HYMAP
HYMAP floodplain roughness:	HYMAP
HYMAP flow direction x:	HYMAP
HYMAP flow direction y:	HYMAP
HYMAP grid elevation:	HYMAP
HYMAP grid distance:	HYMAP
HYMAP grid area:	HYMAP
HYMAP runoff time delay:	HYMAP
HYMAP runoff time delay multiplier:	HYMAP
HYMAP baseflow time delay:	HYMAP
HYMAP basin mask:	HYMAP

Parameter or variable units The unit for the parameter or variable is specified here. If no units, put a “-”.

Time steps or vertical levels specifies the number of timesteps (e.g., 12 for monthly) or number of vertical levels or depths (e.g., 3 soil layer depths). Note: Time and vertical dimensions will be separated in future versions of this LDT input file. Some example values could include:

Value	Description
1	1 time point (static) or 1 vertical layer
12	12 climatological months

Number of types or bins specifies the number types or bins for vegetation, soils or other continuous variables to perform tiling in LIS. Note: Only certain parameters can be greater than 1. Some example values could include:

Value	Description
1	1 bin or 1 value per gridcell
3	Can indicate 3 gridcell fractions for a dataset

Descriptive name for the variable A longer description or name for the variable or parameter is specified here.

```
# VarName, On=1/Off=0, Source, Unit, Vlevs, NumBins, FullName
Landcover: 1 AVHRR - 1 14 "UMD land cover"
Landmask: 1 AVHRR - 1 1 "UMD land mask"
Regional mask: 0 ESRI - 1 1 "State FIPS map"
Lake depth: 0 FLake m 1 1 "Lake depth map"
Lake depth QC: 0 FLake - 1 1 "Lake depth QC file"
Lake wind fetch: 0 FLake m 1 1 "Lake wind fetch"
Lake sediment depth: 0 FLake m 1 1 "Thermally active layer of bottom sediment"
Lake sediment temp: 0 FLake K 1 1 "Thermally active layer temp. of bottom sediment"
Crop type: 0 UMDCROPMAP - 1 32 "UMD+CROPMAP crop types"
Soil texture: 0 STATSGOFAO_LIS - 1 16 "STATSGO+FAO soil texture"
Soil fractions: 0 FAO - 1 1 "FAO sand fraction"
Soil color: 0 FAO - 1 1 "FAO soil color"
Elevation: 0 SRTM_LIS m 1 1 "SRTM elevation"
Slope: 0 SRTM_LIS - 1 1 "SRTM slope"
Aspect: 0 SRTM_LIS - 1 1 "SRTM aspect"
Curvature: 0 GTOP030_LIS - 1 1 "GTOP030 curvature"
Slope type: 0 NCEP_LIS - 1 1 "NCEP LIS-6 slope type"
LAI: 0 AVHRR - 12 1 "AVHRR LAI climatology"
LAImin: 1 CLSMF2.5 - 1 1 "CLSM F2.5 Min LAI climatology"
LAImax: 1 CLSMF2.5 - 1 1 "CLSM F2.5 Max LAI climatology"
SAI: 0 AVHRR - 12 1 "AVHRR SAI climatology"
Albedo: 0 NCEP_LIS - 12 1 "NCEP LIS-6 albedo climatology"
Max snow albedo: 0 NCEP_LIS - 1 1 "NCEP LIS-6 max snow free albedo"
Albedo NIR factors: 0 CLSMF2.5 - 12 1 "CLSM F2.5 alb near-IR scale factors"
Albedo VIS factors: 0 CLSMF2.5 - 12 1 "CLSM F2.5 alb visible scale factors"
Greenness: 0 NCEP_LIS - 12 1 "NCEP LIS-6 greenness climatology"
Shdmin: 0 NCEP_LIS - 1 1 "NCEP LIS-6 greenness min"
Shdmax: 0 NCEP_LIS - 1 1 "NCEP LIS-6 greenness max"
PET: 0 SACHTET.3.5.6 - 12 1 "Monthly PET climatology"
PETADJ: 0 SACHTET.3.5.6 - 12 1 "Monthly adjusted PET climatology"
Porosity: 0 FAO - 1 3 "Porosity"
Quartz: 0 - - 1 1 "Quartz"
Psisat: 0 CLSMF2.5 - 1 1 "Psisat: Clapp-Hornberger parameter (CLSM)"
Bexp: 0 CLSMF2.5 - 1 1 "Bexp: Clapp-Hornberger parameter (CLSM)"
Ksat: 0 CLSMF2.5 ms-1 1 1 "Saturated hydraulic conductivity (CLSM)"
Wpwet: 0 CLSMF2.5 - 1 1 "Wilting point wetness (CLSM)"
Soildepth: 0 ALMIPPII - 1 1 "Soildepth"
Rootdepth: 0 ALMIPPII - 1 1 "Rootdepth"
Bedrockdepth: 0 CLSMF2.5 mm 1 1 "Depth to bedrock (CLSM)"
Tbot: 0 NCEP_LIS K 1 1 "LIS-6 Bottom temperature"
CLSM gnu param: 0 CLSMF2.5 m-1 1 1 "Catchment F2.5 vertical transm. decay term"
```

CLSM ars params:	0	CLSMF2.5	m2kg-1	1	1	"Catchment F2.5 wetness parameters"
CLSM ara params:	0	CLSMF2.5	m2kg-1	1	1	"Catchment F2.5 topographic shape parameters"
CLSM arw params:	0	CLSMF2.5	m2kg-1	1	1	"Catchment F2.5 minimum theta parameters"
CLSM bf params:	0	CLSMF2.5	kgm-4	1	1	"Catchment F2.5 baseflow topographic parameters"
CLSM ts params:	0	CLSMF2.5	-	1	1	"Catchment F2.5 water transfer parameters"
CLSM tau params:	0	CLSMF2.5	-	1	1	"Catchment F2.5 topographic tau parameters"
SAC-HTET:	0	SACHTET.3.5.6	-	1	1	"SAC-HTET parameters"
SNOW-17:	0	SNOW-17	-	1	1	"SNOW17 parameters"
ADC:	0	SNOW-17	-	1	11	"Points on Areal (snow) Depletion Curve"
SiB2:	0	SiB2	-	1	12	"SiB2 parameters"
WRSI LGP:	0	GeoWRSI.2	dekad	1	1	"GeoWRSI 2 Length of growing period"
WRSI WHC:	0	GeoWRSI.2	mm	1	1	"GeoWRSI 2 Water holding capacity"
WRSI SOSclim:	0	GeoWRSI.2	dekad	1	1	"GeoWRSI 2 SOS climatology"
WRSI WRSIclim:	0	GeoWRSI.2	-	1	1	"GeoWRSI 2 WRSI climatology"
WRSI mask:	0	GeoWRSI.2	-	1	1	"GeoWRSI 2 regional land mask"
WRSI SOS:	0	GeoWRSI.2	dekad	1	1	"GeoWRSI 2 SOS (run-time)"
WRSI SOSanom:	0	GeoWRSI.2	dekad	1	1	"GeoWRSI 2 SOS anomaly"
PPT climatology:	0	PRISM	mm	12	1	"PRISM PPT climatology fields"
Irrig area fraction:	0	MODIS	%	1	1	"MODIS Irrig gridcell fraction"
Irrigation type:	0	MODIS	-	1	1	"MODIS Irrigation type map"
HYMAP river width:	0	HYMAP	-	1	1	"HYMAP river width map"
HYMAP river height:	0	HYMAP	-	1	1	"HYMAP river height map"
HYMAP river roughness:	0	HYMAP	-	1	1	"HYMAP river roughness map"
HYMAP river length:	0	HYMAP	-	1	1	"HYMAP river length map"
HYMAP floodplain height:	0	HYMAP	-	1	10	"HYMAP floodplain height map"
HYMAP floodplain roughness:	0	HYMAP	-	1	1	"HYMAP floodplain roughness map"
HYMAP flow direction x:	0	HYMAP	-	1	1	"HYMAP flow direction x map"
HYMAP flow direction y:	0	HYMAP	-	1	1	"HYMAP flow direction y map"
HYMAP grid elevation:	0	HYMAP	-	1	1	"HYMAP grid elevation map"
HYMAP grid distance:	0	HYMAP	-	1	1	"HYMAP grid distance map"
HYMAP grid area:	0	HYMAP	-	1	1	"HYMAP grid area"
HYMAP runoff time delay:	0	HYMAP	-	1	1	"HYMAP runoff time delay map"
HYMAP runoff time delay multiplier:	0	HYMAP	-	1	1	"HYMAP runoff time delay multiplier map"
HYMAP baseflow time delay:	0	HYMAP	-	1	1	"HYMAP baseflow time delay map"
HYMAP basin mask:	0	HYMAP	-	1	1	"HYMAP basin mask map"

A Description of output files from LDT

This section provides a description of various output files generated during an LDT processing run. The main output format for LDT is NetCDF (*.nc). This includes NetCDF with HDF5 compression capabilities.

The output file could be named something like, lis_input.d01.nc. To view the header and/or data information, you will need ncview and/or ncdump utilities, both provided by the Unidata webpage var(<http://www.unidata.ucar.edu/software/netcdf/>).

Some of the main components (as found in the header information) will include:

- dimensions : *east-west, north-south, month, sfctypes, etc.*
- variables : *time, LANDMASK, LANDCOVER, SURFACETYPE, etc.*
- global attributes : *title, references, MAP_PROJECTION, DX, DY, etc.*

A.0.1 Dimensions attributes

The LDT output file, like the NetCDF file, contains header information for the dimensions. The list can include:

```
dimensions:  
    east_west = 80 ;  
    north_south = 37 ;  
    month = 12 ;  
    time = 1 ;  
    sfctypes = 14 ;  
    soiltypes = 16 ;  
    soilfracbins = 3 ;  
    elevbins = 1 ;  
    slopebins = 1 ;  
    aspectbins = 1 ;  
....  
....
```

This file can be used to determine the number of tiles used in a LIS model simulation. The number of tiles are specified in this dimensions header information.

A.0.2 Variable attributes

The LDT output file, like the NetCDF file, contains header information for the variable or parameter file attributes. The list can include: (for example)

```
variables:  
    float LANDCOVER(sfctypes, north_south, east_west) ;  
    LANDCOVER:standard_name = "UMD land cover" ;
```

```
LANDCOVER:units = "-" ;
LANDCOVER:scale_factor = 1.f ;
LANDCOVER:add_offset = 0.f ;
LANDCOVER:missing_value = -9999.f ;
LANDCOVER:vmin = 0.f ;
LANDCOVER:vmax = 0.f ;
LANDCOVER:num_bins = 14 ;
.....
....
```

B Cylindrical Lat/Lon Domain Example

This section describes how to compute the values for the run domain and/or the domain for a parameter or variable file on a cylindrical lat/lon projection.

First, you want to set the bounding coordinates for your desired LIS-based run domain.

If you wish to run over the whole domain defined by a parameter data domain then you simply set the values defined in the parameter domain section in the run domain section. This gives:

```
Run domain lower left lat: -59.875
Run domain lower left lon: -179.875
Run domain upper right lat: 89.875
Run domain upper right lon: 179.875
Run domain resolution dx: 0.25
Run domain resolution dy: 0.25
```

Just note that if you wish to run on a LIS run domain that happens to be greater (e.g. all of North America) than the extents of a read-in parameter file (e.g., STATSGO domain), then checks are in place for LDT to stop the running process.

To process a subsetted LIS-based run domain, here are example extents that can be set:

```
Run domain lower left lat: 27.875
Run domain lower left lon: -97.625
Run domain upper right lat: 31.875
Run domain upper right lon: -92.875
Run domain resolution dx: 0.25
Run domain resolution dy: 0.25
```

See Figure 1 for an illustration of adjusting the running grid. See Figures 2 and 3 for an illustration of the south-west and north-east grid-cells.

LDT is designed to bring a read-in parameter file directly to a common LIS-based domain grid, projection and resolution. Currently, the user can upscale or downscale from a given lat/lon projection to any other lat/lon projection but also lambert conformal and other projections.

The 'LIS-produced' parameter data is defined on a Latitude/Longitude grid, from -180 to 180 degrees longitude and from -60 to 90 degrees latitude. Most 'Native' parameter datasets can extend down to -90 degrees latitude, accounting for glacial areas like Antarctica.

For this example, consider reading in a parameter file at $1/4$ deg resolution. The coordinates of the south-west and the north-east points are specified at the grid-cells' centers. Here the south-west grid-cell is given by the box $(-180, -60), (-179.750, -59.750)$. The center of this box is $(-179.875, -59.875)$.

¹

¹Note, these coordinates are ordered (longitude, latitude).

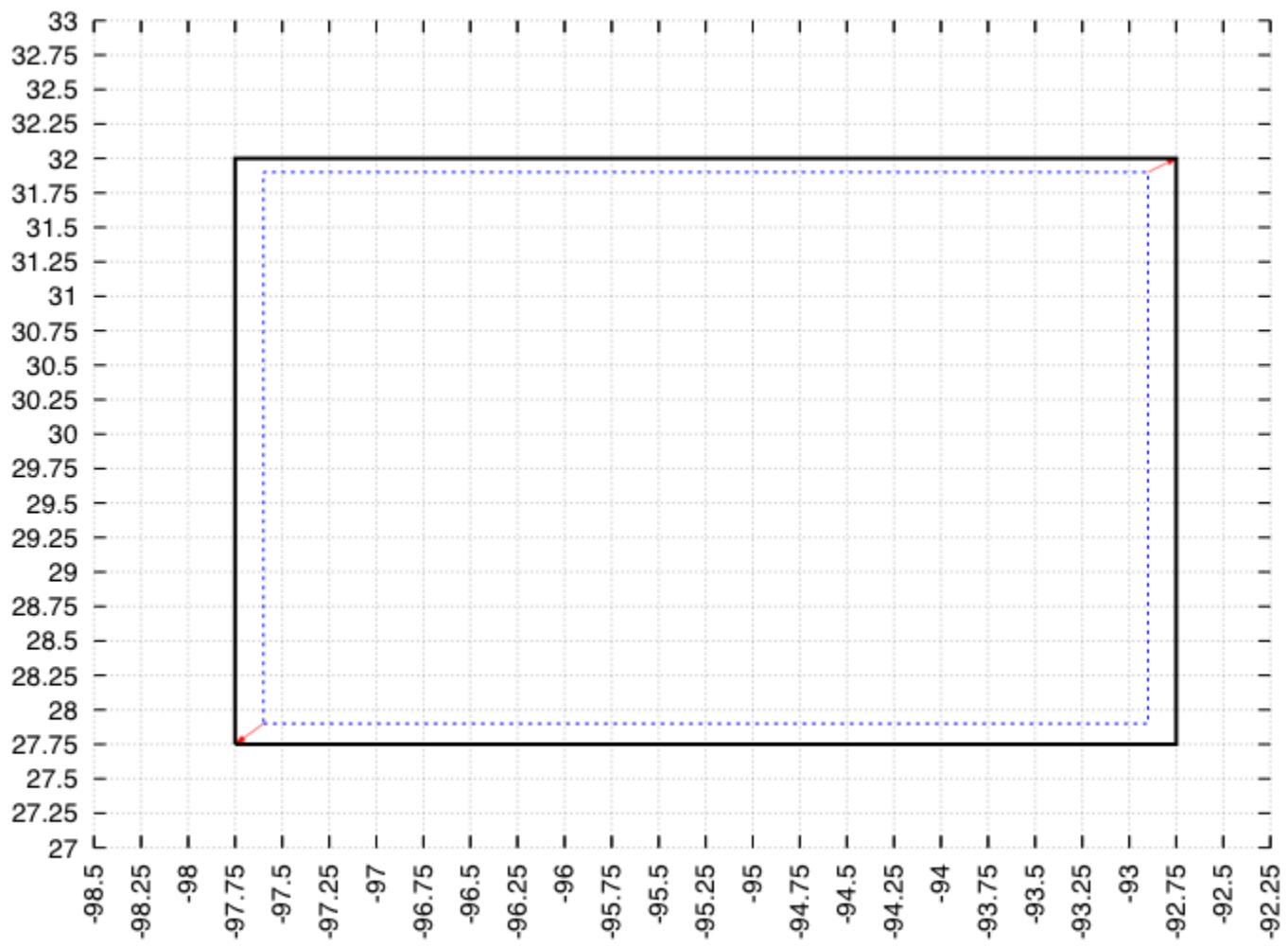


Figure 1: Illustration showing how to fit the desired running grid onto the actual grid

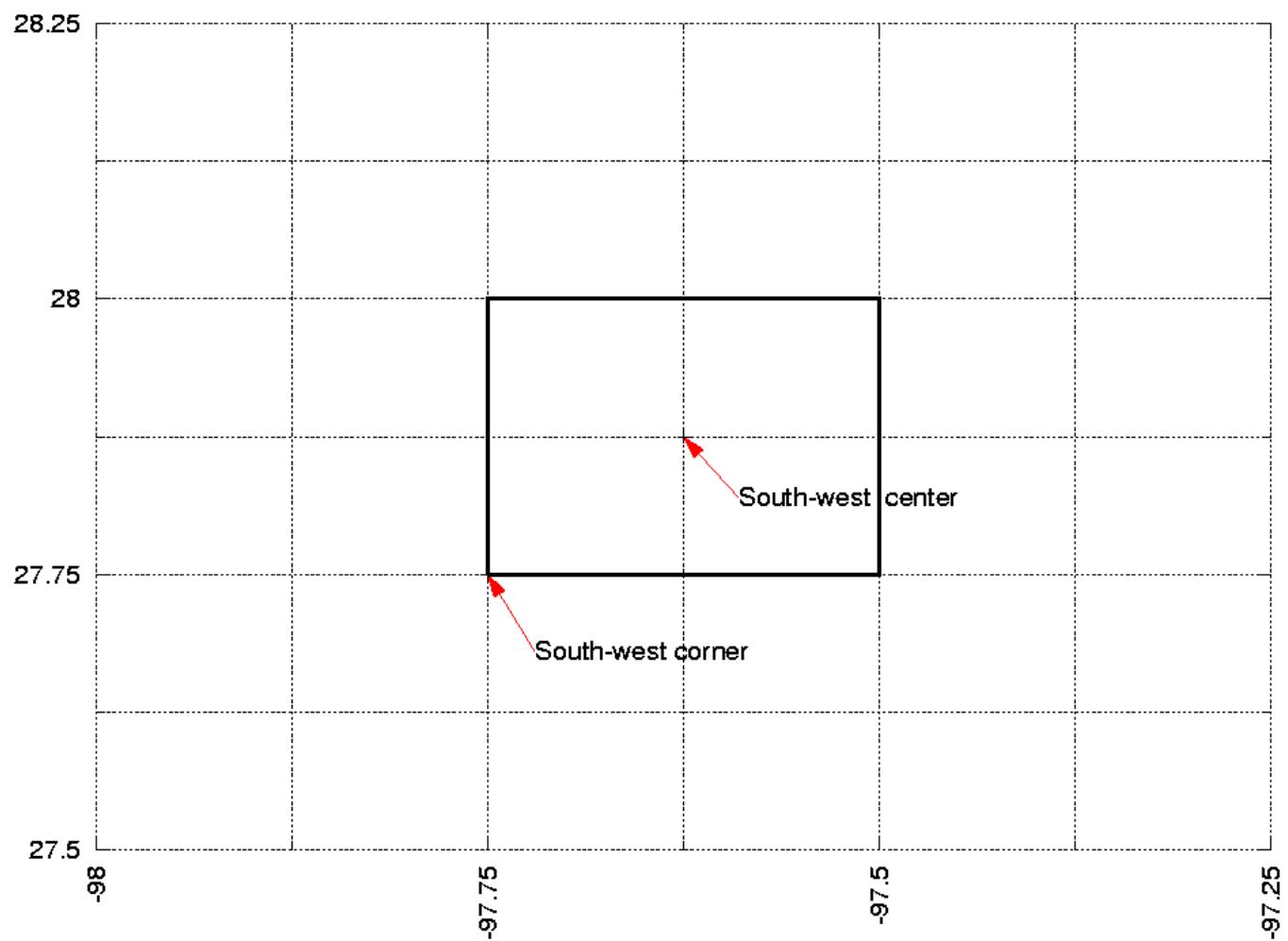


Figure 2: Illustration showing the south-west grid-cell corresponding to the example in Section B

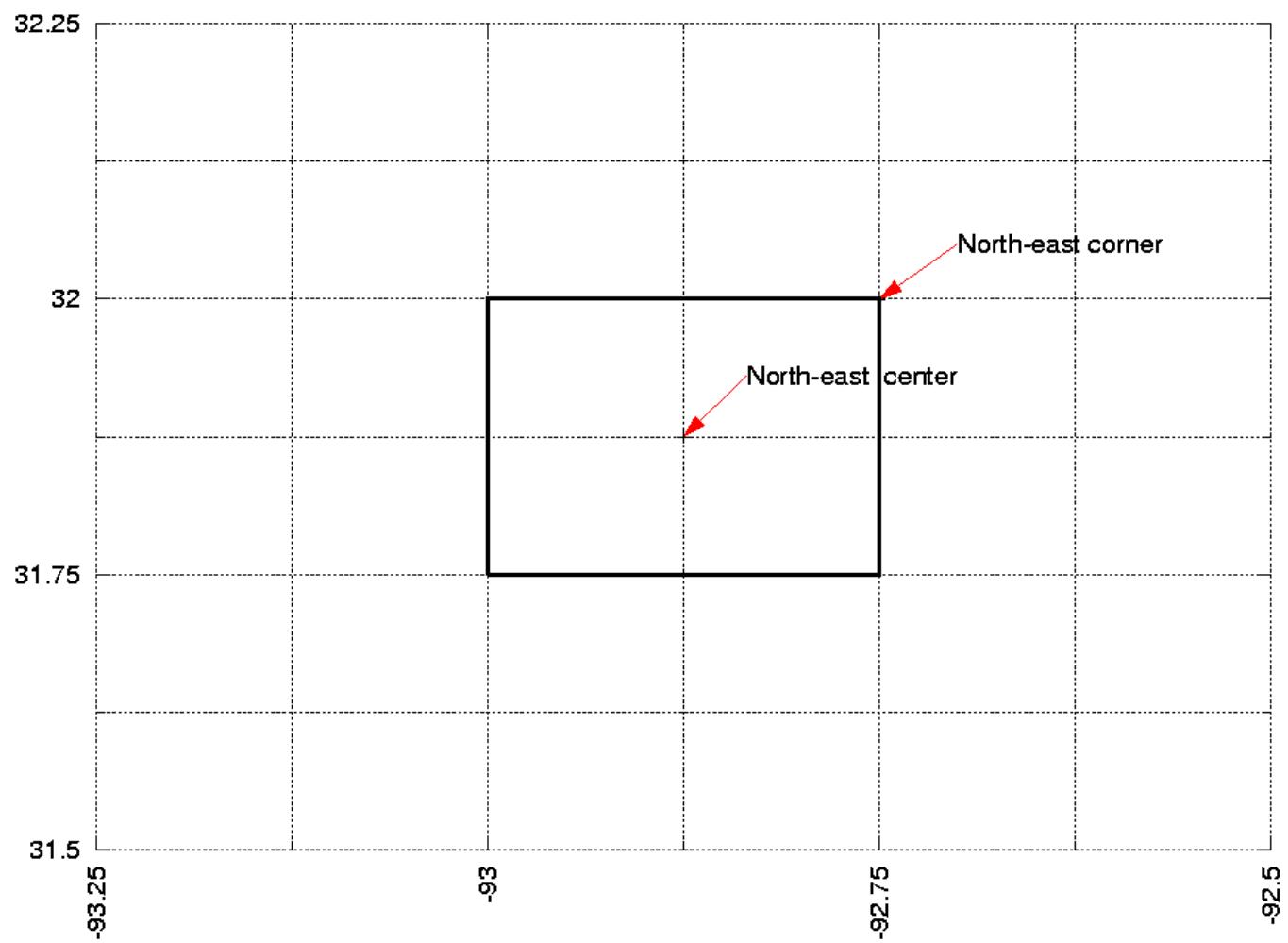


Figure 3: Illustration showing the north-east grid-cell corresponding to the example in Section B

```
param domain lower left lat: -59.875  
param domain lower left lon: -179.875
```

The north-east grid-cell is given by the box $(179.750, 89.750), (180, 90)$. Its center is $(179.875, 89.875)$.

```
param domain upper right lat: 89.875  
param domain upper right lon: 179.875
```

Setting the resolution (0.25 deg) gives

```
param domain resolution dx: 0.25  
param domain resolution dy: 0.25
```

And this completely defines a parameter data domain example.

C Lambert Conformal Domain Example

This section describes how to compute the values for the run domain and param domain sections on a Lambert conformal projection.

Note that this projection is often used for a coupled run with the Weather Research and Forecasting (WRF) model. As such, Lambert domains are first generated when configuring WRF via WRF's preprocessing system (WPS). The domain information is then copied into LIS' *lis.config* file.

Please see WRF's User's Guide found at <http://www.mmm.ucar.edu/wrf/users/publicdoc.html> for more information.

D Gaussian Domain Example

This section describes how to compute the values for the run domain and param domain sections on a Gaussian projection.

If you wish to run over the whole domain defined by the parameter data domain then you simply set the values defined in the parameter domain section in the run domain section. This gives:

Run domain first grid point lat:	-89.27665
Run domain first grid point lon:	0.0
Run domain last grid point lat:	89.27665
Run domain last grid point lon:	-0.9375
Run domain resolution dlon:	0.9375
Run domain number of lat circles:	95

If you wish to run over a sub-domain, then you must choose longitude and latitude values that correspond to the T126 Gaussian projection. Tables of acceptable longitude and latitude values are found below.

Now say you wish to run only over the region given by $(-97.6, 27.9), (-92.9, 31.9)$. Since the running domain must fit on the T126 Gaussian grid, the running domain must be expanded to $(-98.4375, 27.87391), (-91.875, 32.59830)$. Thus the running domain specification is:

Run domain first grid point lat:	27.87391
Run domain first grid point lon:	-98.4375
Run domain last grid point lat:	32.59830
Run domain last grid point lon:	-91.875
Run domain resolution dlon:	0.9375
Run domain number of lat circles:	95

Table 1: Acceptable longitude values

0.000000	0.937500	1.875000	2.812500	3.750000
4.687500	5.625000	6.562500	7.500000	8.437500
9.375000	10.312500	11.250000	12.187500	13.125000
14.062500	15.000000	15.937500	16.875000	17.812500
18.750000	19.687500	20.625000	21.562500	22.500000
23.437500	24.375000	25.312500	26.250000	27.187500
28.125000	29.062500	30.000000	30.937500	31.875000
32.812500	33.750000	34.687500	35.625000	36.562500
37.500000	38.437500	39.375000	40.312500	41.250000
42.187500	43.125000	44.062500	45.000000	45.937500
46.875000	47.812500	48.750000	49.687500	50.625000
51.562500	52.500000	53.437500	54.375000	55.312500
56.250000	57.187500	58.125000	59.062500	60.000000
60.937500	61.875000	62.812500	63.750000	64.687500
65.625000	66.562500	67.500000	68.437500	69.375000
70.312500	71.250000	72.187500	73.125000	74.062500
75.000000	75.937500	76.875000	77.812500	78.750000
79.687500	80.625000	81.562500	82.500000	83.437500
84.375000	85.312500	86.250000	87.187500	88.125000
89.062500	90.000000	90.937500	91.875000	92.812500
93.750000	94.687500	95.625000	96.562500	97.500000
98.437500	99.375000	100.312500	101.250000	102.187500
103.125000	104.062500	105.000000	105.937500	106.875000
107.812500	108.750000	109.687500	110.625000	111.562500
112.500000	113.437500	114.375000	115.312500	116.250000
117.187500	118.125000	119.062500	120.000000	120.937500
121.875000	122.812500	123.750000	124.687500	125.625000
126.562500	127.500000	128.437500	129.375000	130.312500
131.250000	132.187500	133.125000	134.062500	135.000000
135.937500	136.875000	137.812500	138.750000	139.687500
140.625000	141.562500	142.500000	143.437500	144.375000
145.312500	146.250000	147.187500	148.125000	149.062500
150.000000	150.937500	151.875000	152.812500	153.750000
154.687500	155.625000	156.562500	157.500000	158.437500
159.375000	160.312500	161.250000	162.187500	163.125000
164.062500	165.000000	165.937500	166.875000	167.812500
168.750000	169.687500	170.625000	171.562500	172.500000
173.437500	174.375000	175.312500	176.250000	177.187500
178.125000	179.062500	180.000000	-179.062500	-178.125000

-177.187500	-176.250000	-175.312500	-174.375000	-173.437500
-172.500000	-171.562500	-170.625000	-169.687500	-168.750000
-167.812500	-166.875000	-165.937500	-165.000000	-164.062500
-163.125000	-162.187500	-161.250000	-160.312500	-159.375000
-158.437500	-157.500000	-156.562500	-155.625000	-154.687500
-153.750000	-152.812500	-151.875000	-150.937500	-150.000000
-149.062500	-148.125000	-147.187500	-146.250000	-145.312500
-144.375000	-143.437500	-142.500000	-141.562500	-140.625000
-139.687500	-138.750000	-137.812500	-136.875000	-135.937500
-135.000000	-134.062500	-133.125000	-132.187500	-131.250000
-130.312500	-129.375000	-128.437500	-127.500000	-126.562500
-125.625000	-124.687500	-123.750000	-122.812500	-121.875000
-120.937500	-120.000000	-119.062500	-118.125000	-117.187500
-116.250000	-115.312500	-114.375000	-113.437500	-112.500000
-111.562500	-110.625000	-109.687500	-108.750000	-107.812500
-106.875000	-105.937500	-105.000000	-104.062500	-103.125000
-102.187500	-101.250000	-100.312500	-99.375000	-98.437500
-97.500000	-96.562500	-95.625000	-94.687500	-93.750000
-92.812500	-91.875000	-90.937500	-90.000000	-89.062500
-88.125000	-87.187500	-86.250000	-85.312500	-84.375000
-83.437500	-82.500000	-81.562500	-80.625000	-79.687500
-78.750000	-77.812500	-76.875000	-75.937500	-75.000000
-74.062500	-73.125000	-72.187500	-71.250000	-70.312500
-69.375000	-68.437500	-67.500000	-66.562500	-65.625000
-64.687500	-63.750000	-62.812500	-61.875000	-60.937500
-60.000000	-59.062500	-58.125000	-57.187500	-56.250000
-55.312500	-54.375000	-53.437500	-52.500000	-51.562500
-50.625000	-49.687500	-48.750000	-47.812500	-46.875000
-45.937500	-45.000000	-44.062500	-43.125000	-42.187500
-41.250000	-40.312500	-39.375000	-38.437500	-37.500000
-36.562500	-35.625000	-34.687500	-33.750000	-32.812500
-31.875000	-30.937500	-30.000000	-29.062500	-28.125000
-27.187500	-26.250000	-25.312500	-24.375000	-23.437500
-22.500000	-21.562500	-20.625000	-19.687500	-18.750000
-17.812500	-16.875000	-15.937500	-15.000000	-14.062500
-13.125000	-12.187500	-11.250000	-10.312500	-9.375000
-8.437500	-7.500000	-6.562500	-5.625000	-4.687500
-3.750000	-2.812500	-1.875000	-0.937500	

Table 2: Acceptable latitude values

-89.27665	-88.33975	-87.39729	-86.45353	-85.50930
-84.56487	-83.62028	-82.67562	-81.73093	-80.78618
-79.84142	-78.89662	-77.95183	-77.00701	-76.06219
-75.11736	-74.17252	-73.22769	-72.28285	-71.33799
-70.39314	-69.44830	-68.50343	-67.55857	-66.61371
-65.66885	-64.72399	-63.77912	-62.83426	-61.88939
-60.94452	-59.99965	-59.05478	-58.10991	-57.16505
-56.22018	-55.27531	-54.33043	-53.38556	-52.44069
-51.49581	-50.55094	-49.60606	-48.66119	-47.71632
-46.77144	-45.82657	-44.88169	-43.93681	-42.99194
-42.04707	-41.10219	-40.15731	-39.21244	-38.26756
-37.32268	-36.37781	-35.43293	-34.48805	-33.54317
-32.59830	-31.65342	-30.70854	-29.76366	-28.81879
-27.87391	-26.92903	-25.98415	-25.03928	-24.09440
-23.14952	-22.20464	-21.25977	-20.31489	-19.37001
-18.42513	-17.48025	-16.53537	-15.59050	-14.64562
-13.70074	-12.75586	-11.81098	-10.86610	-9.921225
-8.976346	-8.031467	-7.086589	-6.141711	-5.196832
-4.251954	-3.307075	-2.362196	-1.417318	-0.4724393
0.4724393	1.417318	2.362196	3.307075	4.251954
5.196832	6.141711	7.086589	8.031467	8.976346
9.921225	10.86610	11.81098	12.75586	13.70074
14.64562	15.59050	16.53537	17.48025	18.42513
19.37001	20.31489	21.25977	22.20464	23.14952
24.09440	25.03928	25.98415	26.92903	27.87391
28.81879	29.76366	30.70854	31.65342	32.59830
33.54317	34.48805	35.43293	36.37781	37.32268
38.26756	39.21244	40.15731	41.10219	42.04707
42.99194	43.93681	44.88169	45.82657	46.77144
47.71632	48.66119	49.60606	50.55094	51.49581
52.44069	53.38556	54.33043	55.27531	56.22018
57.16505	58.10991	59.05478	59.99965	60.94452
61.88939	62.83426	63.77912	64.72399	65.66885
66.61371	67.55857	68.50343	69.44830	70.39314
71.33799	72.28285	73.22769	74.17252	75.11736
76.06219	77.00701	77.95183	78.89662	79.84142
80.78618	81.73093	82.67562	83.62028	84.56487
85.50930	86.45353	87.39729	88.33975	89.27665

E Polar Stereographic Domain Example

This section describes how to compute the values for the run domain and param domain sections on a polar stereographic projection.

STUB!

F HRAP Domain Example

This section describes how to compute the values for the run domain and param domain sections on a HRAP projection.

STUB!

The spatial HRAP resolution has a default value of 1.0, which indicates a horizontal resolution of 4.7625 KM (valid at 1.0). Finer resolution values are then entered as 0.5, 0.25, etc. to represent half, quarter, etc. of the original 4.7625 KM gridcell length. All other HRAP-based polar stereographic grid parameters are already handled within the LDT or LIS code, since HRAP utilizes specific true latitude, standard longitude, and orientation values.

Note that HRAP is a special case of a polar stereographic grid. For HRAP,
true lat = 60.0
standard lon = -105.000
orientation = 0.0
resolution at true lat is 4.7625 km when resolution is set to 1.0; i.e.,

Run domain hrap resolution: 1

G Mercator Domain Example

This section describes how to compute the values for the run domain and param domain sections on a Mercator projection.

Note that this projection is often used for a coupled run with the Weather Research and Forecasting (WRF) model. As such, Mercator domains are first generated when configuring WRF via WRF's preprocessing system (WPS). The domain information is then copied into LIS' *lis.config* file.

Please see WRF's User's Guide found at <http://www.mmm.ucar.edu/wrf/users/pub-doc.html> for more information.

References

- [1] W. Sawyer and A. da Silva. Protex: A sample fortran 90 source code documentation system. Technical report, NASA GMAO, 1997. DAO Office Note 97-11.